

THE GCARS SYSTEM
FORTRAN IV PROGRAMMERS MANUAL
PART B - PROGRAMS FOR GENERATION OF
ALTERNATIVES

SEPTEMBER 1969 - NUMBER 26



BY

A.K. TURNER

JHRP

JOINT HIGHWAY RESEARCH PROJECT
PURDUE UNIVERSITY AND
INDIANA STATE HIGHWAY COMMISSION



Final Report (Part III-B)

THE GCARS SYSTEM FORTRAN IV PROGRAMMERS

MANUAL PART B - PROGRAMS FOR

GENERATION OF ALTERNATIVES

To: J. F. McLaughlin, Director
Joint Highway Research Project

September 1969

File: 1-6-1

From: H. L. Michael, Associate Director
Joint Highway Research Project

Project: C-36-72A

Attached is Part III-B of the Final Report on the research project "Evaluation of Numerical Surface Techniques Applied to Highway Location Analysis" which was approved on July 1, 1967. This report has been authored by Dr. A. Keith Turner of our staff.

This report describes in detail and documents the FORTRAN computer programs used by the GCARS System to generate a series of alternative highway locations in terms of one or more highway location factors, alone or in combinations. It is one of a series of four reports documenting the various phases of the GCARS System.

This report is presented to the Board for review and comment and for the record.

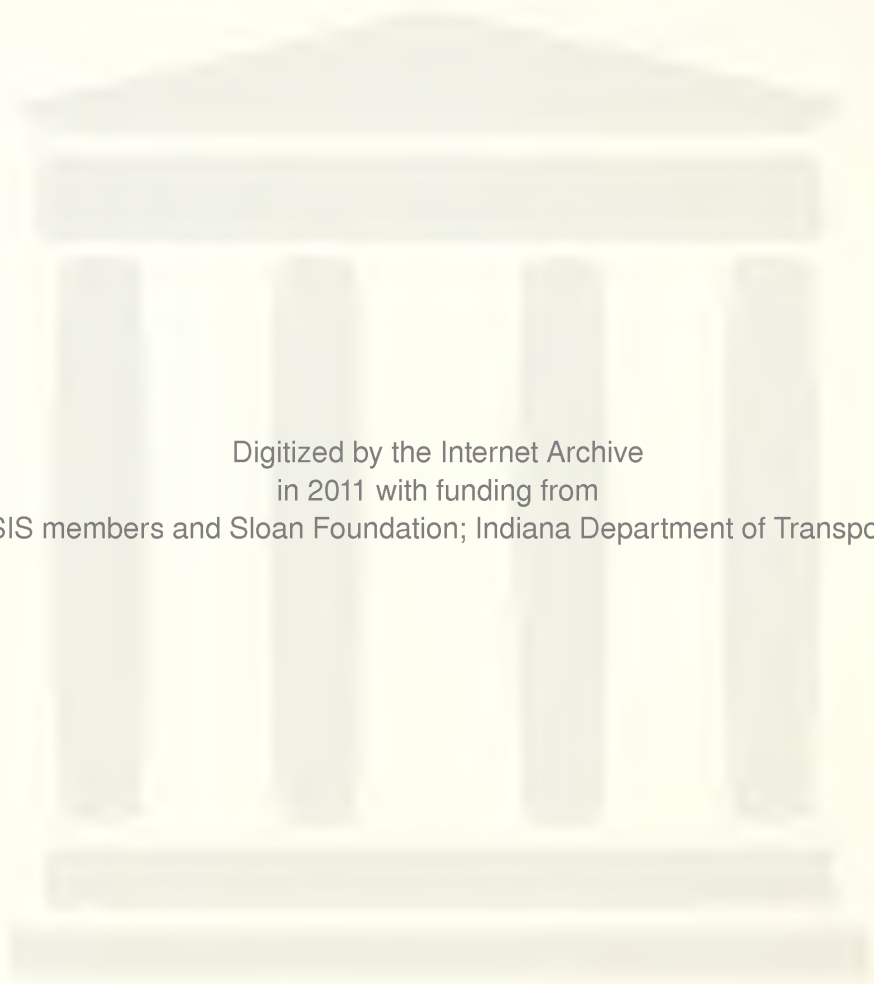
Respectfully submitted,

Harold L. Michael /HLM

Harold L. Michael
Associate Director

HLM:mz

cc: F. L. Ashbaucher	R. H. Harrell	C. F. Scholer
W. L. Dolch	J. A. Havers	M. B. Scott
W. H. Goetz	V. E. Harvey	W. T. Spencer
W. L. Grecco	G. A. Leonards	H. R. J. Walsh
G. K. Hallock	F. B. Mendenhall	K. B. Woods
M. E. Harr	R. D. Miles	E. J. Yoder



Digitized by the Internet Archive
in 2011 with funding from
LYRASIS members and Sloan Foundation; Indiana Department of Transportation

Final Report (Part III-B)

THE GCARS SYSTEM FORTRAN IV PROGRAMMERS MANUAL

PART B - PROGRAMS FOR GENERATION OF ALTERNATIVES

by

A. Keith Turner
Post-Doctoral Research Instructor

Joint Highway Research Project

Project: C-36-72A

File: 1-6-1

Prepared as Part of an Investigation

Conducted by

Joint Highway Research Project
Engineering Experiment Station
Purdue University

in cooperation with the

Indiana State Highway Commission

and the

U.S. Department of Transportation
Federal Highway Administration
Bureau of Public Roads

The opinions, findings and conclusions expressed in this publication are those of the authors and not necessarily those of the Bureau of Public Roads

Not Released for Publication

Subject to Change

Not Reviewed By
Indiana State Highway Commission
or the
Bureau of Public Roads

Purdue University
Lafayette, Indiana
September 1969

N O T I C E

These programs have been tested and are believed to be free of errors. If errors are located, the Joint Highway Research Project would appreciate notification.

Send comments to:-

Joint Highway Research Project
ATTN: Professor R. D. Miles
School of Civil Engineering
Purdue University
Lafayette, Indiana 47907

TABLE OF CONTENTS

	<u>Page</u>
LIST OF FIGURES	ii
ACKNOWLEDGMENTS	iii
INTRODUCTION	B- 1
METHOD OF OPERATION	B- 2
PRESENT CAPACITIES, OPTIONS, AND RESTRICTIONS	B- 4
a) Size of Value Matrices	B- 4
b) Number of Links in Network	B- 4
c) Spacing of Network	B- 4
d) Number of Factors Included in One Analysis	B- 4
e) Input Options	B- 5
f) Output Options	B- 5
FUTURE MODIFICATIONS	B- 5
METHOD OF COMPUTATION	B- 6
a) Program NETSUM	B- 6
b) Subroutine MPATH	B-11
c) Subroutine PATHS	B-20
PROGRAM OPERATING CHARACTERISTICS	B-23
a) Core Storage Required	B-24
b) Running Time	B-24
c) Control Cards	B-24
PROGRAM LISTINGS AND DOCUMENTATION	B-24
a) The use of FORSTAT Data	B-26
b) Listing of Program NETSUM	B-28
c) FORSTAT Cross-Reference Listings for Program NETSUM	B-32
d) Listing of Subroutine MPATH	B-35
e) FORSTAT Cross-Reference Listings for Subroutine MPATH	B-37
f) Listing of Subroutine PRTOUT	B-39
g) FORSTAT Cross-Reference Listings for Subroutine PRTOUT	B-42
h) Listing of Subroutine PATHS	B-44
i) FORSTAT Cross-Reference Listings for Subroutine PATHS	B-47
j) Example of Control Cards and Data for Program NETSUM	B-49
k) Example Output	B-53
l) Example of Punched Output	B-57
REFERENCES	B-59

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	GCARS System Programs for Generation of Alternatives.	B-3
2	Flowchart for Program Netsum.	B-7
3	Method of Defining Origin and Destination of Alternatives.	B-10
4	Link Development Sequence from any Node "N".	B-10
5	Simple Utility Matrix and the Resulting Utility Network Used in Example Calculations.	B-12
6	Link Table for Network in Figure 5.	B-13
7	The Cumulative and Tree Tables for the Network in Figure 5.	B-14
8	Development of Partial Tree to Discover Minimum Path from 12 to 1 in Sample Network.	B-15
9	Flowchart for Subroutine MPAATH.	B-16
10	Parameters Used in Developing a Printer-Map of Generated Alternatives.	B-22
11	Typical Card Deck Setup for One Analysis.	B-25

ACKNOWLEDGMENTS

The author is grateful to the Joint Highway Research Project of Purdue University, the Indiana State Highway Commission and the Bureau of Public Roads for their financial support.

The author wishes to express his sincere appreciation to Mr. John Steele, Associate Director, Purdue Computer Science Center, and his staff, without whose aid the documentation of these programs could not have been completed in the time available. Thanks are also due to Miss M. Wong, Mr. R. Herrick, and Mr. T. Lassen for drafting the illustrations and assembling and checking the computer print-outs utilized in this report.

Professor Robert D. Miles, P. E., directed the project and reviewed the manuscript. His assistance is gratefully acknowledged.

THE GCARS SYSTEM
FORTRAN IV PROGRAMMERS MANUAL
PART B - PROGRAMS FOR GENERATION OF ALTERNATIVES

By
A. Keith Turner

INTRODUCTION

The GCARS System generates a series of alternative highway alignments by analyzing a series of "route-independent" highway location factors. Basic information on each factor is supplied as a "value surface" representing the potential costs or benefits associated with the particular factor. The programs concerned with the development and testing of these value surfaces are described in Part A of this manual (5).*

The generation of alternatives involves the following four steps (3):-

- (1) Convert value matrices to utility matrices (value surfaces are always described in a matrix form within the computer).
- (2) If multiple factors are being analyzed, combine the appropriate utility matrices, each multiplied by a selected weighting factor, to produce a combined utility matrix.
- (3) Develop a network of links (a utility network) joining adjacent modes on the utility matrix so that routes can be selected across the area.
- (4) Generate a series of alternatives by repeated minimum path analysis to the network.

This portion of the Programmers Manual describes the computer routines developed to perform these steps. The routines are:

* Number in parenthesis refer to references.

Program NETSUM - Accepts value matrices, converts them to utility matrices, and combines them in specified ratios to produce a combined utility matrix. Constructs a utility network link table suitable for the minimum path algorithm.

Subroutine MPATH - Performs the repeated minimum path analysis to generate a specified number of alternatives.

Subroutine PRTOUT - Output routine, prepares printed output for NETSUM and MPATH.

Subroutine PATHS -- Produces a map of the generated alternatives on the printer.

The following two routines, described in Part C of this manual, are used in conjunction with the above routines:

Subroutine STATS - (use required) - called by NETSUM during the transformation of values to utilities, and also by PRTRCON4.

Subroutine PRTRCON4 - (use optional) - may be called by NETSUM to produce a printer-contour map of the combined utility surface.

METHOD OF OPERATION

Figure 1 is a simplified flow chart showing the transfer of control between the routines during the analysis.

Program NETSUM exerts the primary control. All control cards, containing option specifications defined by the engineer, are read by this program. It then calls on the associated subroutines as required.

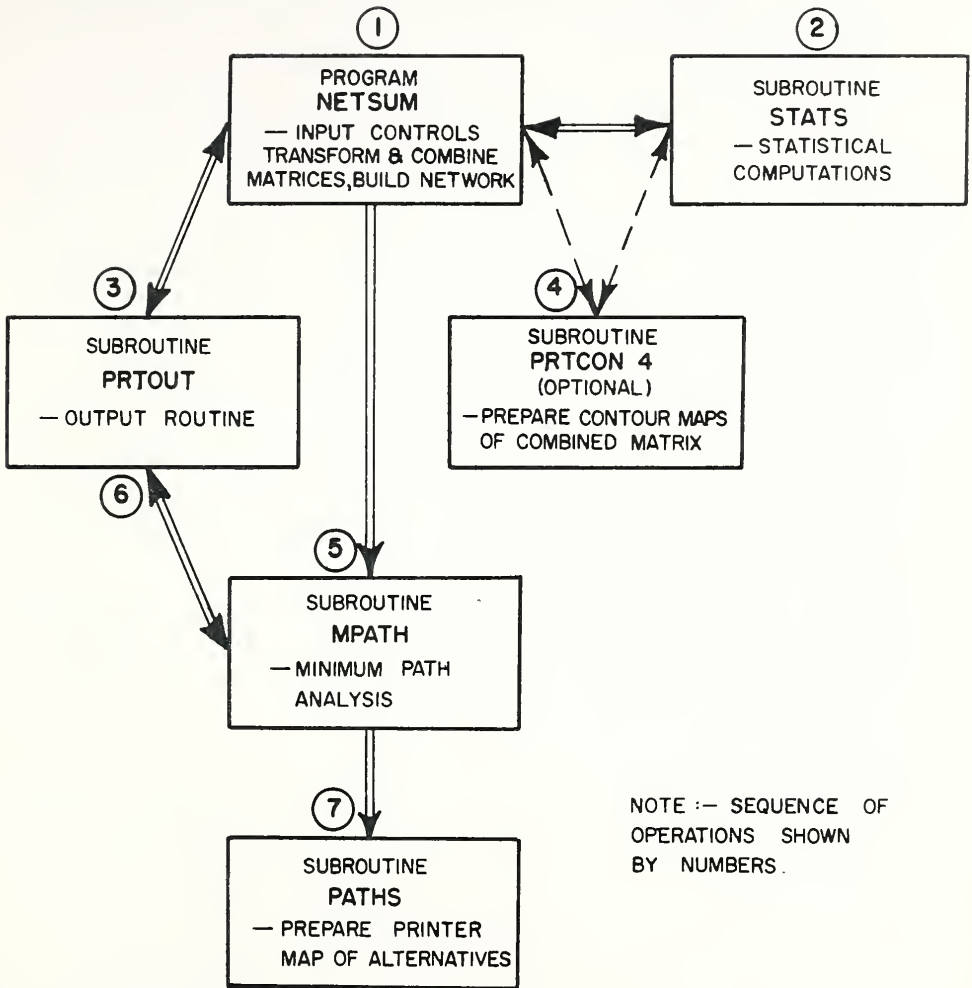


FIGURE 1 . GCARS SYSTEM PROGRAMS FOR GENERATION OF ALTERNATIVES.

PRESENT CAPACITIES, OPTIONS, AND RESTRICTIONS

a) Size of Value Matrices

The present maximum size of value matrices is 50 x 50. Neither dimension can exceed these limits without correcting the DIMENSION statements in the various programs. Changes in shape, for instance to 100 x 25, can be accomplished without increasing the overall storage requirements of the programs. An overall increase in size will require a complete redimensioning of most variables.

Special Note: The value matrices (Z, ZZ) are dimensioned
Z(NROW, NCOL).

where NROW = number of rows (height of matrix)

NCOL = number of columns (width of matrix)

b) Number of Links in Network

The maximum number is 9800. This corresponds to the number required by a 50 x 50 matrix with four links out of every central nodes, three out of edge nodes, and two out of corner nodes.

c) Spacing of Network

The present networks are assumed to be composed of square grids - the spacing between columns equals the spacing between rows. This can be modified by changing values of XDIST and/or YDIST in NETSUM (lines 18 and 19 page B-30) to reflect the relative spacings of columns and rows. Further corrections would be required to correct the maps produced by PRTPCON⁴ and PATHS.

d) Number of Factors Included in One Analysis

The present maximum number is 16. This can be changed merely by redimensioning VAL in Program NETSUM.

e) Input Options

The default option is card decks. However an optional request allows the value surfaces and their leader cards to be read from standard GCARS binary tapes. The characteristics of such tapes are described in the Users Manual (4). A suitable tape can be developed by Program LOADG (5).

f) Output Options

Standard output includes (1) table of factors and weights, (2) table of alternatives showing path totals, relative path values, and number of links per path (see page B-55), and (3) map of paths of alternatives.

Debug option gives additional output, but tends to destroy neatness of some standard output tables. Additional items include (1) listings of input value matrices, (2) listings of transformed utility matrices, (3) listing of combined utility matrix, (4) link summaries for all paths.

Punch option produces punched deck of combined utility matrix (see page B-57).

Contours option produces printer contour map of combined utility matrix. Various contour displays are possible, for details read the section describing Subroutine PRICON4 in Part C of this manual (6).

FUTURE MODIFICATIONS

The following are eight options which might be considered in applying GCARS to actual location problems. Many of these modifications or options require only a small amount of new programming. The following items might be changed:-

- (1) Size and/or shape of value matrices.
- (2) Shape of grid (at present a "square grid is assumed).

- (3) Central percentage of links to be removed from a developed path before generation of next alternative (at present 93% of links removed).
- (4) Optional complete removal of certain links or nodes from a network to completely prevent travel over certain subareas.
- (5) Retain links in network over certain control sections so that many choices can utilize a particular bridge or tunnel crossing site.
- (6) Increase the maximum number of choices to be generated (present maximum = 7).
- (7) Utilize table of nodes passed through by each alternative to search and tabulate values of various factors along each link of each choice.
- (8) Revise method of weighting topographic residuals so that cuts are no longer rated equal to fills, so that deep cuts or fills are weighted more expensive per foot increase after an initial shallow height or depth.

METHOD OF COMPUTATION

a) Program NETSUM

Figure 2 is a flowchart for Program NETSUM. After reading the control cards, NETSUM reads a GCARS header card and the value matrix (Z) from either tape or cards. The variable ITYPE on the header card specifies the type of transformation that will be applied to convert the values to utilities.

If ITYPE = 1, residual values are to be minimized. A residual of -2.0, representing a fill condition, is weighted equal to a residual of +2.0, representing a cut of equal depth. This is accomplished by

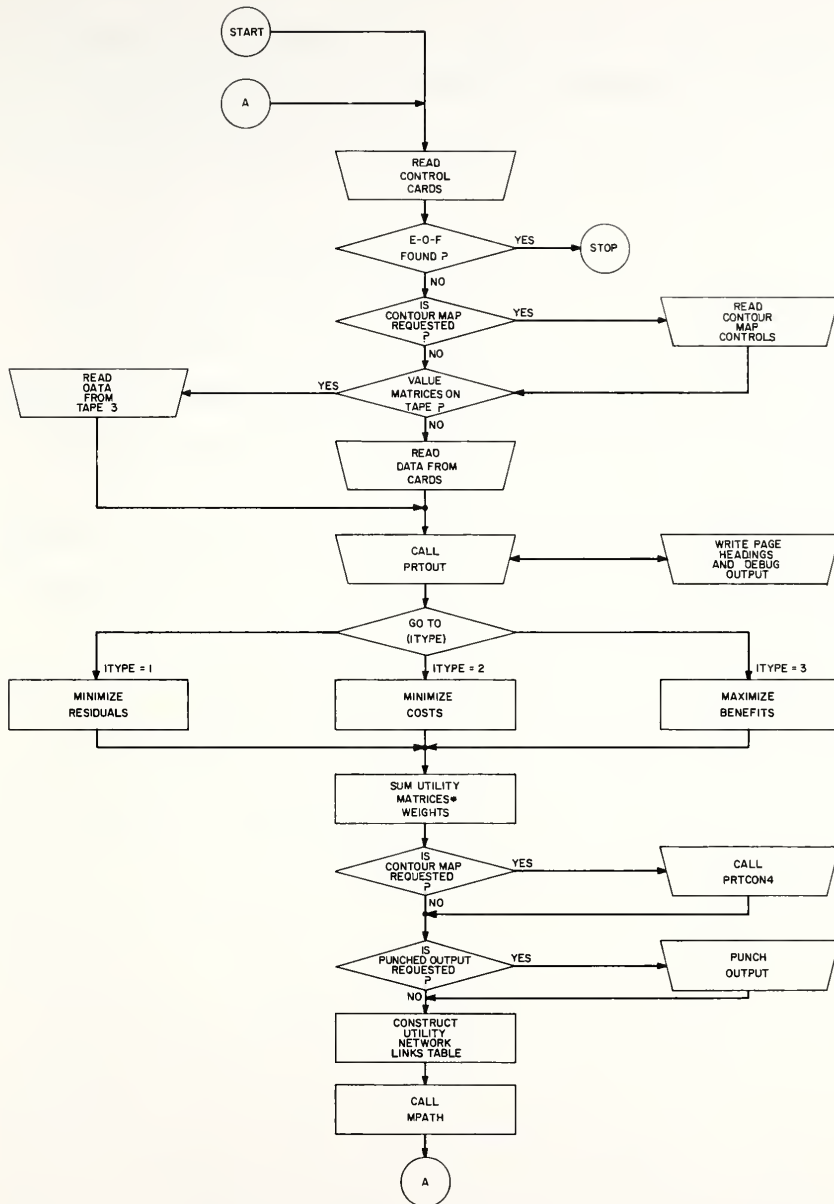


FIGURE 2. FLOWCHART FOR PROGRAM NETSUM.

setting all values of the Z matrix to their absolute values; $Z(I,J) = \text{ABS}(Z(I,J))$. Then Subroutine STATS is called to determine the maximum and minimum values, these are used to standardize the Z matrix values between 0.0 and 10.0 by:

$$Z(I,J) = ((Z(I,J) - \text{MINZ})/(\text{MAXZ}-\text{MINZ}))*10.0.$$

If ITYPE = 2, the values are in terms of costs which only need standardizing to a range of 0.0 to 10.0. This is accomplished, as above, by calling Subroutine STATS to determine the maximum and minimum values, then dividing each matrix value by the range and multiplying the result by ten.

If ITYPE = 3, the values are in terms of benefits and must therefore be standardized then subtracted from ten to invert the "benefits" to "costs".

$$Z(K,J) = 10.0 - ((Z(I,J)-\text{MINZ})/\text{MAXZ}-\text{MINZ}))*10.0$$

The resulting utility matrix is then multiplied by its appropriate weighting factor (VAL(N)) and summed with all preceding utility-weight products to produce the combined utility matrix (ZZ).

$$\text{ZZ}(I,J) = Z(I,J)*\text{VAL}(N) + \text{ZZ}(I,J)$$

In the above equations:

N = number of factor in sequence of factors

I = Row coordinate of element

J = column coordinate of element

Z = utility matrix for this factor

ZZ = combined utility matrix (initially set = 0.0)

When completely generated, the ZZ matrix is transferred to the Z matrix locations ($Z(I,J) = \text{ZZ}(I,J)$) and the ZZ locations are released for other uses.

NETSUM allows the origin and destination nodes of the alternatives to be entered by row and column coordinates. These must be converted for the minimum path algorithm to a single quantity as follows:

$$\text{origin} = \text{NHOME} = (\text{IX}-1) * \text{NCOL} + \text{IY}$$

$$\text{destination} = \text{NDEST} = (\text{JX}-1) * \text{NCOL} + \text{JY}$$

Figure 3 defines these variables.

The link table is next constructed from the combined utility matrix elements. The value of each link (TLINK) is the average of the utilities of its two end points. The links are ordered in the table according to their origin nodes; the first origin node is the top left-hand corner element (Z(1,1)), the last origin node is the bottom right hand corner (Z(NROW,NCOL)).

Central nodes in the network have four links radiating from them. These are computed and stored in the order shown in Figure 4.

Nodes in the four corners of the network can have only two connecting links, those on the sides only three. However on large matrices the link to node ratio approaches four. As adjacent nodes are used in sequence, a network is slowly completed that includes links allowing travel from N1 to N2 and N2 to N1, in other words, an "undirected" network (3).

The minimum path algorithm requires three pieces of information for each link:--(1) its origin node (N), (2) its destination node (J), and (3) its utility value (TLINK). The original version of the algorithm thus required three storage locations for each link and for large networks this rapidly filled the available core memory on the CDC 6500 computer.

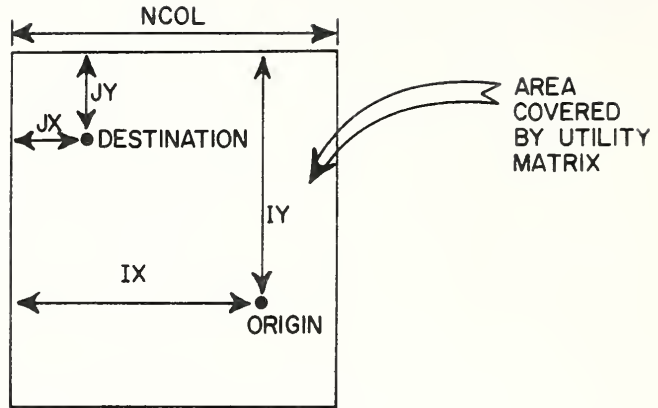


FIGURE 3. METHOD OF DEFINING ORIGIN AND DESTINATION OF ALTERNATIVES.

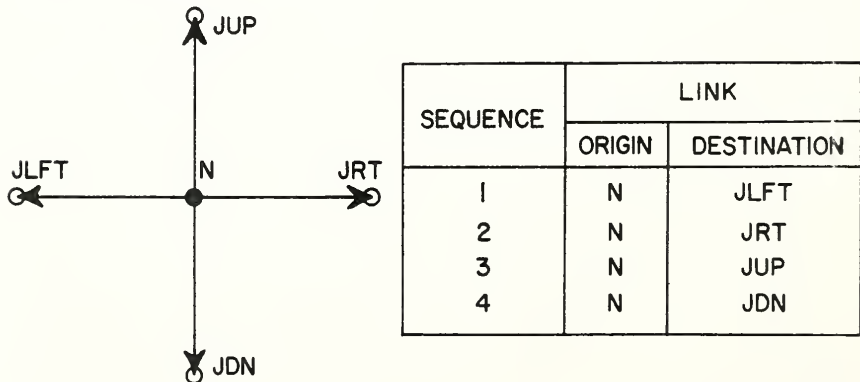


FIGURE 4. LINK DEVELOPMENT SEQUENCE FROM ANY ORIGIN NODE "N".

The storage requirements were therefore reduced by "packing" the origin and destination information into a single number called NXJ according to the simple formula:

$$NXJ = N * IPACK + J$$

where NXJ = combined origin-destination code.

N = origin node

IPACK = a very large constant number

J = destination node

The selected value of IPACK was 40000_8 whose decimal equivalent is 16384. This value is large enough to prevent computational problems in networks joining 2500 nodes (i.e. 50 x 50 matrix), but should be increased if the network dimensions are increased. The NXJ values are "unpacked" by Subroutine MPATH to obtain the appropriate N and J values, through the use of integer arithmetic as follows:

$$N = NXJ / IPACK$$

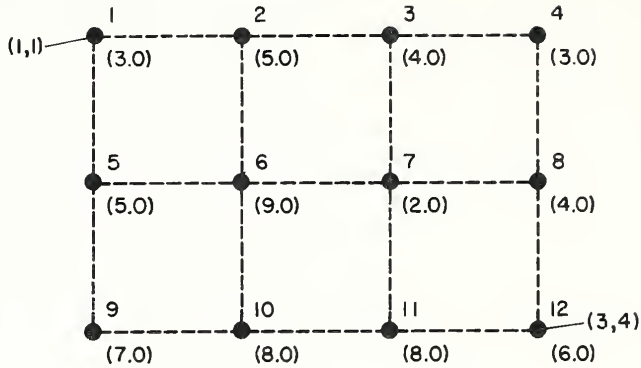
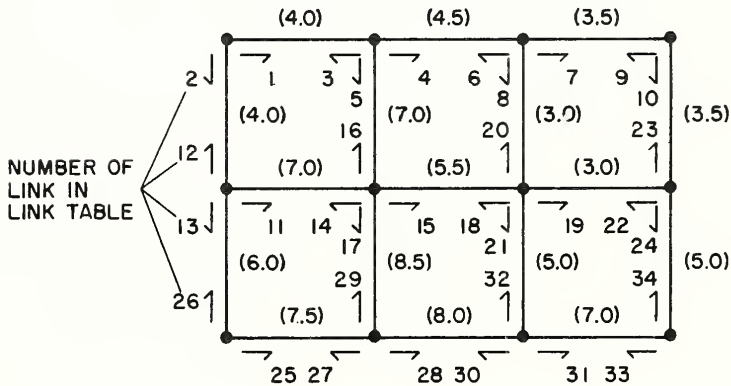
$$J = NXJ - N * IPACK$$

Figure 5 shows a sample utility matrix, and the computed utility network. The resulting link table is shown in Figure 6.

b) Subroutine MPATH

Subroutine MPATH utilizes a minimum path algorithm developed by the British Road Research Laboratory (8). This RRL algorithm was coded in FORTRAN by Martin (2). MPATH utilizes a modified version of the Martin coding; the modifications are summarized in a following section.

Figure 9 is a flowchart summarizing the operation of this algorithm. Three tables are used in a computation, a TREE table, a CUMULATIVE table, and the prepared LINK table. The computations required to determine the minimum path from node 12 to node 1 on the sample network in Figure 5

A - ORIGINAL UTILITY MATRIXB - RESULTING UTILITY NETWORK

(NUMBERS IN BRACKETS ARE UTILITIES OF NODES (A) OR LINKS (B))

$$\text{LINK/NODE RATIO} = 34/12 = 2.38$$

FIGURE 5. SIMPLE UTILITY MATRIX AND THE RESULTING UTILITY NETWORK USED IN EXAMPLE CALCULATIONS.

I	N	J	NXJ	TLINK
1	1	2	16386	4.0
2	1	5	16389	4.0
3	2	1	32769	4.0
4	2	3	32771	4.5
5	2	6	32774	7.0
6	3	2	49154	4.5
7	3	4	49156	3.5
8	3	7	49160	3.0
9	4	3	65539	3.5
10	4	8	65544	3.5
11	5	6	81926	7.0
12	5	1	81921	4.0
13	5	9	81929	6.0
14	6	5	98309	7.0
15	6	7	98311	5.5
16	6	2	98306	7.0
17	6	10	98314	8.5
18	7	6	114694	5.5
19	7	8	114696	3.0
20	7	3	114691	3.0
21	7	11	114699	5.0
22	8	7	131079	3.0
23	8	4	131076	3.5
24	8	12	131084	5.0
25	9	10	147466	7.5
26	9	5	147461	6.0
27	10	9	163849	7.5
28	10	11	163851	8.0
29	10	6	163846	8.5
30	11	10	180234	8.0
31	11	12	180236	7.0
32	11	7	180231	5.0
33	12	11	196619	7.0
34	12	8	196616	5.0

FIGURE 6. THE LINK TABLE FOR NETWORK IN
FIGURE 5.

CUMULATIVE TABLE

ORDER OF ENTRY	NCUM	TCUM	ORDER OF REMOVAL	ENTRIES FOUND IN LINK TABLE BUT DISCARDED BECAUSE END NODE ALREADY REACHED.
1	33	7.0	2	
2	34	5.0	1	
3	22	8.0	3	
4	23	8.5	4	24
5	30	15.0	9	31
6	32	12.0	6*	
7	18	13.5	8	19
8	20	11.0	5	21
9	9	12.0	7*	10
10	6	15.5	10	
11	14	20.5		15
12	16	20.5		
13	17	22.0		
14	27	22.5		28,29
15	3	19.5	11	4,5

*These entries found to be repetitive, thus not placed in TREE Table.

TREE TABLE

ORDER OF ENTRY	J	NN	TSUM	
9	1	Ø 3	~~99999.99~~ 19.5	NDEST
8	2	Ø 6	~~99999.99~~ 15.5	
5	3	Ø 20	~~99999.99~~ 11.0	
4	4	Ø 23	~~99999.99~~ 8.5	
	5	0	99999.99	
6	6	Ø 18	~~99999.99~~ 13.5	
3	7	Ø 22	~~99999.99~~ 8.0	
1	8	Ø 34	~~99999.99~~ 5.0	
	9	0	99999.99	
7	10	Ø 30	~~99999.99~~ 15.0	
2	11	Ø 33	~~99999.99~~ 7.0	NHOME
START	12	0	0.0	

FIGURE 7. THE CUMULATIVE AND TREE TABLES FOR THE NETWORK IN FIGURE 5.

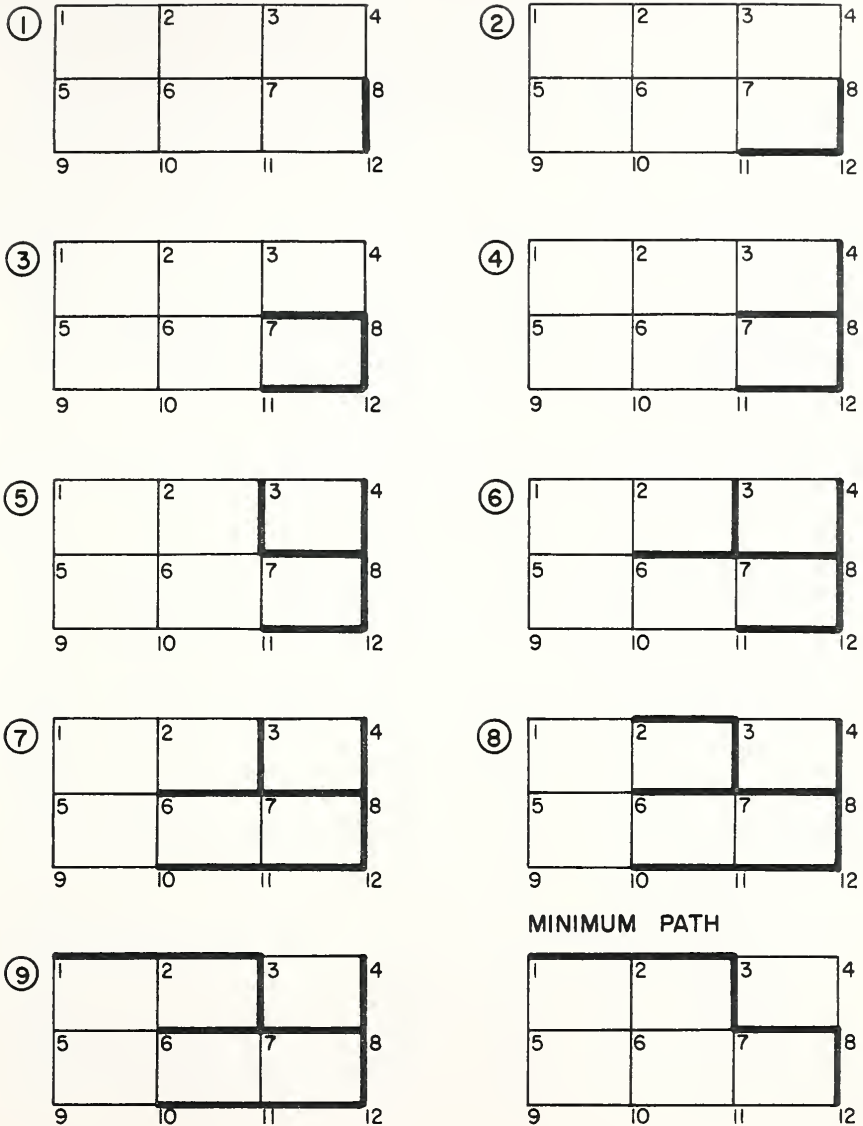


FIGURE 8. DEVELOPMENT OF PARTIAL TREE TO DISCOVER MINIMUM PATH FROM 12 TO 1 IN SAMPLE NETWORK.

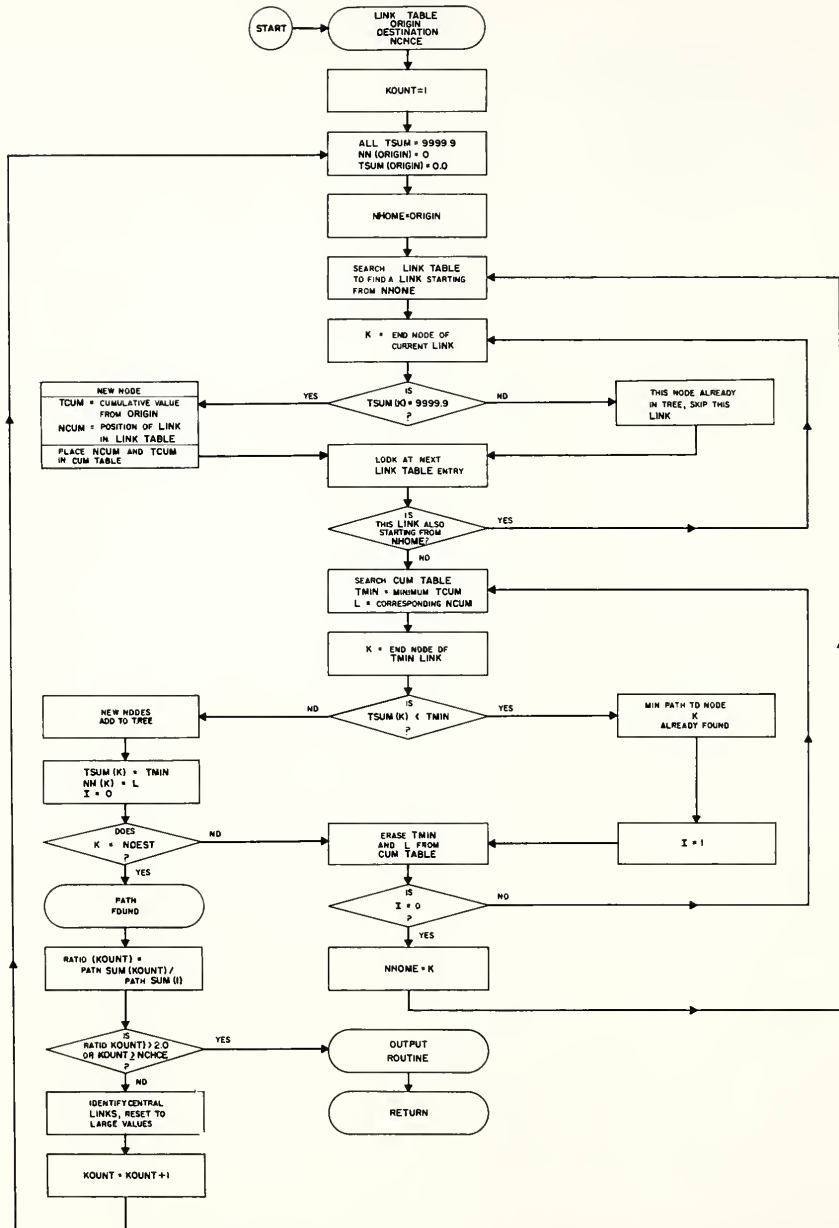


FIGURE 9. FLOWCHART FOR SUBROUTINE MPATH.

are shown in Figure 7. Figure 8 shows the stages in the development of the partial tree.

i) Initialization Procedures:

The NCUM, TCUM and NN entries in the CUMULATIVE AND TREE tables are set to zero. The TSUM values in the TREE table are set to arbitrarily extreme values (99999.99). The TSUM entry for the origin node (NHOME) is now set to zero.

ii) Development of Partial Tree:

The LINK table is searched to find an entry having NHOME as an origin node. A modification was made to the original Martin FORTRAN coding at this point. Martin arranged for the search of the LINK table to begin at the nth entry; where $n = \text{NHOME}$. (In the example, where $\text{NHOME} = 12$, the search would begin at the 12th entry). This will save search time and is always possible since the link/node ratio is greater than one. Note however in the example network (Figure 6) the link/node ratio is 2.833 so that the search starting at entry 12 has to extend to 33 before a link originating at 12 is encountered. If the search is begun at an entry some multiple of NHOME, according to the link/node ratio, the length of search is greatly reduced. Accordingly the algorithm coding has been modified so that the search of the LINK table begins at $n - \text{NHOME} * \text{LNKNDE}$, where LNKNDE is the integer value of the link/node ratio (in the example $\text{LNKNDE} = 2$). The search now begins at entry 24, and lasts only 9 rather than 21, entries. This simple modification speeded up the computation by a factor of two.

The first link originating at NHOME is examined and the destination node J is determined. If the TSUM entry for J is 99999.99, J has not been reached yet by the developing tree. The link is therefore entered into the CUMULATIVE table, TCUM being updated to reflect the cumulative sum of the link values (TLINK) from NHOME to J. If the TSUM entry for J is less than 99999.99, J must have already been reached. Since the algorithm always finds the best path to a node the first time it reaches it, there is no point in considering other links leading to J. Thus they are not added to the CUMULATIVE table. This discarding of repetitive links saves computation time in large problems.

The NCUM value in the CUMULATIVE table is set to reflect the position of the link in the LINK table. It thus acts as a pointer.

The next entry in the LINK table is examined to see if it also originates from NHOME. If it does its destination node becomes J, and it is checked and added to the CUMULATIVE table or discarded as just described. This process is repeated until all the links originating at NHOME are found (in the example there are only two).

The algorithm then picks the minimum TCUM value in the CUMULATIVE table. This is the "cheapest" path from NHOME to node J in line NCUM of the LINK table. Values are then updated in the TREE table; at line J of this table NN is set equal to the line number in the LINK table and TSUM is set equal to TCUM. This minimum TCUM entry is then removed from the CUMULATIVE table, and the node J becomes the new NHOME (see Figure 7).

The entire above process is repeated with this new NHOME, further entries are added to the CUMULATIVE table, and the minimum TCUM is selected as just described. The process continues until the entry being removed has a J which corresponds to NDEST. When it does a path has been found between the desired origin and destination so that the process can stop.

One additional check is made during the operation of the algorithm. As each minimum TCUM is selected from the CUMULATIVE table it is compared with the corresponding TSUM value in the TREE table, to see if it is still less. Although this check was made as entries were being placed in the CUMULATIVE table, it is possible for that particular node to have been reached prior to the removal of this particular link from the CUMULATIVE table. If this has happened the TSUM value will be less than the TCUM value, this TCUM value is therefore eliminated and a new TCUM searched for this occurs twice during the worked example (The 6th and 7th links being removed are marked with a * to show this). The reader is invited to work through the sample calculations himself, to learn the operation of the algorithm.

iii) Development of Multiple Alternatives:

The above calculations will serve to locate only a single path - the "minimum" or "best" path - between any origin-destination node-pair. Further modifications are required to develop multiple, distinct alternatives between an origin and a destination. Ayad considered this problem (1). Accordingly the algorithm was modified to reflect his concepts. The process is quite simple:

- 1) Find the minimum path between the origin and destination. This becomes a "first choice".
- 2) "Remove" a central percentage of the links forming this path from further consideration. Such links can be "removed" by raising their TLINK values to very high values, making them extremely unattractive during further studies.
- 3) Re-apply the minimum path algorithm to the revised network to find a new minimum path. This will be not quite as good as the first minimum path located earlier, and so will become a "second choice".

- 4) Calculate the ratio of the path total of this choice to the path total along the first choice. If this ratio is not too large, "remove" the central links of this choice also from further consideration and return to step (3).
- 5) Repeat the analysis until either enough alternatives have been generated, or until the ratio of the currently generated path sum to the first choice path sum exceeds a pre-determined cut-off value.

The modifications that perform this process are coded into the latter part of Subroutine MPATH. The nodes through which each passes are stored in a matrix called NODE, the number of links in each path are stored in NUMLK. These values are utilized by Subroutine PATHS to generate a map of the alternatives.

iv) Summary of Modifications to the RRL Algorithm:

The most important modifications to the algorithm are:

- (1) changed search procedure in the LINK table, which speeds up the computations by a factor of two.
- (2) the addition of methods to "remove" links from the network and thus generate multiple alternatives.
- (3) the storage of the information concerning the path sums (TOTAL), path ratios (RATIO), number of links/path (NUMLK), and the nodes comprising each path (NODE) so that further manipulations are possible.

c) Subroutine PATHS

Subroutine PATHS produces maps of the generated alternatives on the printer. The routine begins by scaling the map; given an overall width for the final map in inches it computes the maximum number of print

positions (characters) it can assign to each grid across the page, and how many rows down the page. This is accomplished by computing the following:

$$\text{COL} = \text{WIDTH} * 10.0 + 2.0$$

where COL is an integer number and equals the number of characters allowable since there are ten characters/inch.

$$\text{PS} = (\text{COL} - \text{COLS})/(\text{COLS} - 1)$$

where COLS = number of columns in matrix.

PS now is the number of characters between columns (see Figure 10).

The final map width will thus be $\text{COLMAX} = \text{PS} * (\text{COLS} - 1) + \text{COLS}$.

It will never be greater than the requested size. The number of characters per grid is thus $\text{XSTOP} = \text{PS} + 1$.

The number of rows per grid (at 6 rows of type per inch) is thus

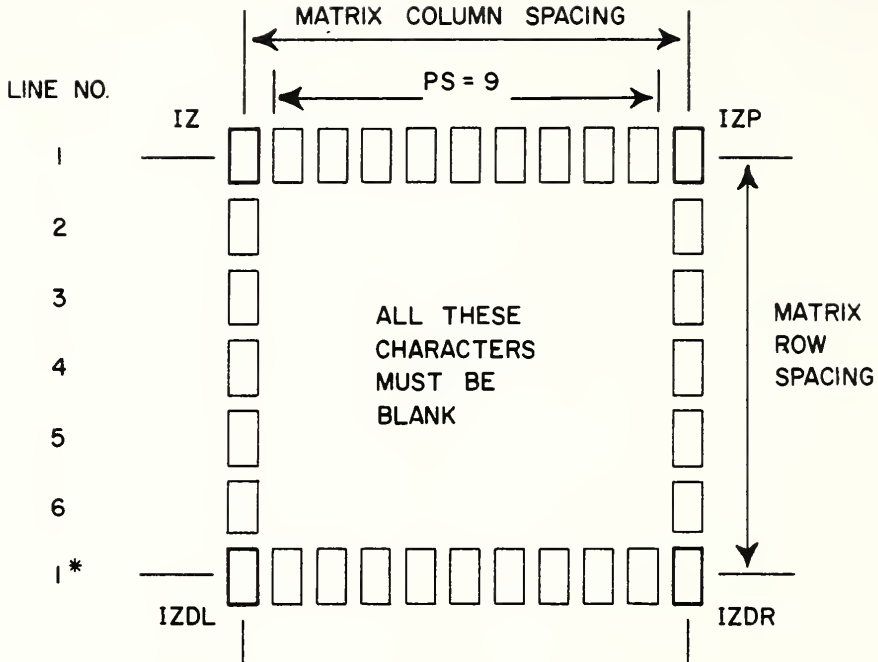
$$\text{YSTOP} = ((\text{PS} + 1) * 6 + 5) / 10$$

All the above calculations are done in integer arithmetic.

The actual map is computed and printed line by line, to save storage requirements. A line of blanks and plus signs is first generated (SYM) to define the column spacing along the top of the map. Once inside the map however only two possibilities exist

- 1) the current line being generated passes through the grid points
or
- 2) the current line is between grid points.

The computations also reflect the fact that within each grid there is one line (the first or top line) that passes through the grid points, followed by a sequence of one or more lines, each of which will be identical, that lie between grid points. This occurs because the paths travel only from grid point to grid point along either vertical or horizontal links and cannot cut across grid squares. Figure 10 shows the



* LINE NO. 1 OF NEXT MATRIX ROW DOWN MAP.

NOTE :— LINE NO.1 IS THE ONLY ONE TO PASS THROUGH MATRIX NODES .

LINES 2-6 MUST BE IDENTICAL.

FIGURE 10. PARAMETERS USED IN DEVELOPING A PRINTER-MAP OF GENERATED ALTERNATIVES.

defined variables. The program begins with a line of blank characters. The program next calculates if the line is a top line through grid points or an intermediate line. If it is a top line it is only necessary to check each path in turn to see if it passes through IZ or IZP. If it passes through either one or the other, the end characters should be set to the appropriate number. If both IZ and IZP fall on the same path, the path must pass from one to the other, and all intermediate characters are set to the path number.

In the case of the second line (the first intermediate line) all four positions IZ, IZP, IZDL and IZDR, must be checked. If IZ and IZDL both lie on the same path, the path must travel up the left edge of the square. If IZP and IZDR both lie on the same path, the path must travel up the right edge of the square. When either of these situations occurs, the appropriate end character is set.

This analysis is repeated for each square along a line and for each choice starting at the worst choice generated (this allows better choices to superimpose on poorer ones). A savings in computation time is effected by noting that the third, fourth, etc. lines are all the same as the second line. Once it is generated, it can be printed the required number of times without recalculation.

This is repeated for the number of rows in the matrix, at which time a map of all the alternatives will have been produced.

PROGRAM OPERATING CHARACTERISTICS

a) Core Storage Required

On the CDC 6500 Program NETSUM and its associated subroutines (including PRTCON4 and excluding HYP50) can be compiled, loaded, and executed in 116K (octal) storage locations when dimensioned to handle

50 x 50 matrices (maximum). The storage requirements will be reduced if the maximum matrix size is less. For instance matrices around 35 x 30 can be handled in just under 60K storage locations.

b) Running Time

This is somewhat data dependent, and thus slightly unpredictable. It is linearly dependent on the number of choices being generated. The average time required to generate five choices for a combination of several factors is about 100 Sec CP-time and 15 Sec PP-time. This includes compiling time as well as the production of a contour map of the combined utility surface, the path statistics, and a map of the alternatives. For the same data, by way of comparison, the program compiled and two choices were generated with similar output in 50 Sec CP-time and 15 Sec PP-time.

c) Control Cards

Figure 11 shows a typical deck set up for the generation of alternatives. The control card parameters are defined in detail on pages B-28,B-29, under the description of NETSUM, since it utilizes all control cards.

The first two control cards are always required. The contour map control cards are needed only when the contour map option is requested. The header cards and value matrices may be omitted if these are to come from tape. Many data sets can be submitted as a single run.

PROGRAM LISTINGS AND DOCUMENTATION

The following pages contain listings of the various routines in order. All are written in basic FORTRAN IV. Each routine is documented through the use of appropriate comment cards at the beginning and throughout the program.

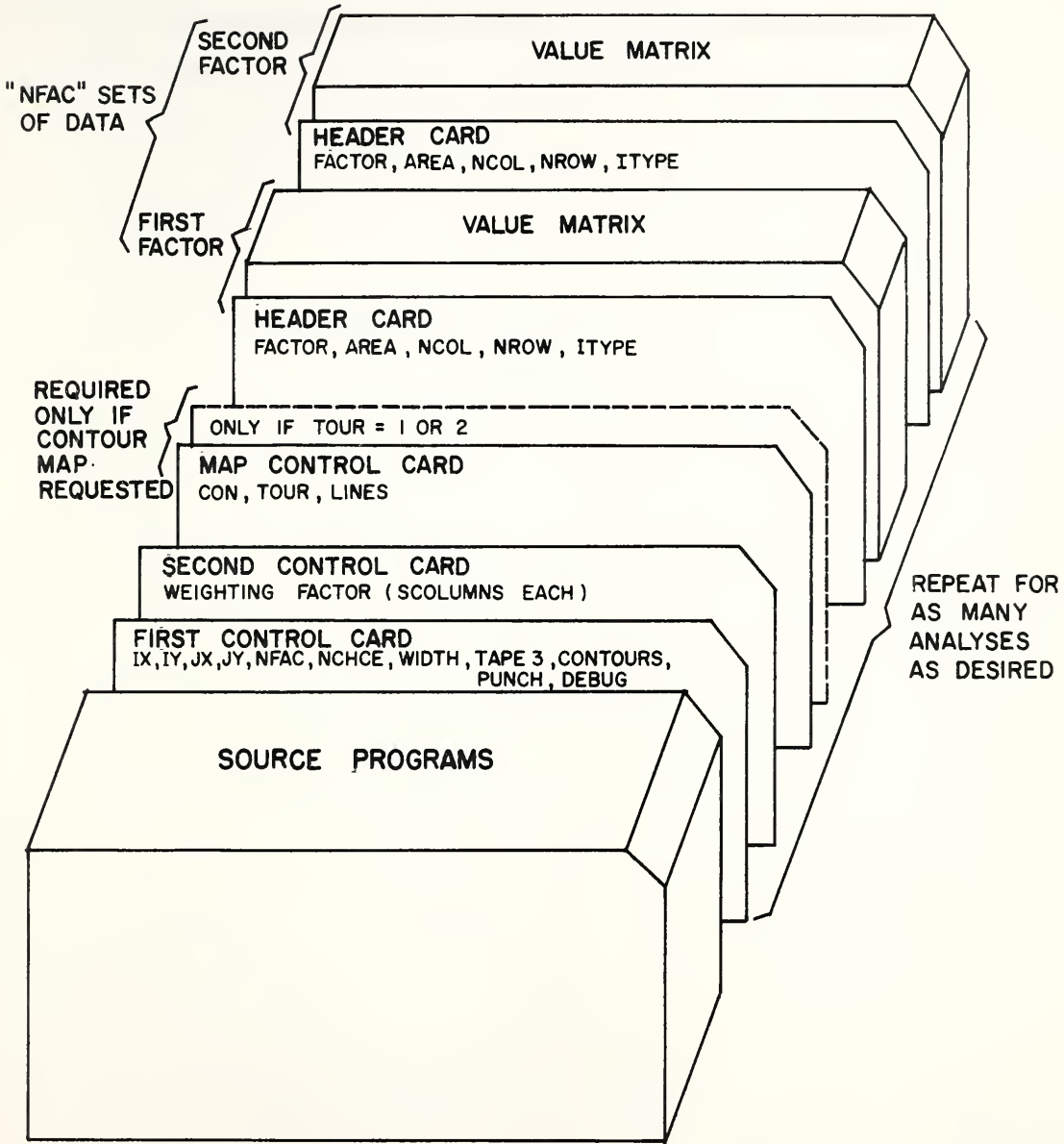


FIGURE II. TYPICAL CARD DECK SETUP FOR ONE ANALYSIS.

Each program has been analyzed by Program TIDY, available as a library program at Purdue, which resequences all statement numbers, removes extraneous numbers, and generally makes the program neater. For this reason the statement numbers in each routine increase progressively in increments of five, and all FORMATS (if any) are collected at the end. The cards are each given a sequence number which appears at the far right.

a) The Use of FORSTAT Data

Each routine was next analyzed by another program, FORSTAT. This checks the syntax of source (FORTRAN) programs and produces cross-reference listings of all variables and statement numbers found within the routine. The cross-reference listings utilize a series of "FORSTAT Numbers" which are found at the far left of the listing. A vertical ink margin line has been added to separate these numbers from the rest of the listing. The cross-reference listings are found at the end of each routine.

All symbols defined in the routine are listed alphabetically and are identified according to type, definition and references (where referenced).

The following symbols are utilized in the cross-reference listings:-

Types

I = INTEGER

R = REAL

D = DOUBLE PRECISION

C = COMPLEX

L = LOGICAL

N = NAME LIST

ASF = ARITHMETIC STATEMENT FUNCTION

Definition

C = COMMON

D = DIMENSION

E = EQUIVALENCE

All other terms are believed self-explanatory. These cross-reference listings will be most useful should any reader desire to make changes to these routines.

b) LISTING OF PROGRAM NETSUM

```

1. PROGRAM NETSUM(INPUT,OUTPUT,TAPE3,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT) 2
PROGRAM NETSUM 4
----- 6
C 8
C 10
C 12
C PURPOSE -- 14
C TO PREPARE VALUE SURFACES FOR GENERATION OF ALTERNATIVES BY 16
C MINIMUM PATH ANALYSIS (SUBROUTINE MPATH). 18
C NETSUM TAKES ONE OR MORE VALUE MATRICES, CONVERTS THEM TO 20
C UTILITY MATRICES BY STANDARDIZATION AND INVERSION (IF NECESSARY). 22
C AND MULTIPLIES THEM BY SUPPLIED WEIGHTING FACTORS TO PRODUCE THE 24
C COMBINED UTILITY SURFACE. 26
C THE COMBINED UTILITY SURFACE IS ANALYZED TO PRODUCE A LINK 28
C TABLE. 30
C ----- 32
C THE GCARS SYSTEM 34
C PURDUE UNIVERSITY 36
C 38
C A. KEITH TURNER, 40
C CIVIL ENGINEERING, 42
C JUNE 1969. 44
C 46
C ----- 48
C 50
C REQUIRED SUBROUTINES-- 52
C -STATS 54
C -PRINTOUT 56
C -MPATH 58
C -PATHS 60
C 62
C OPTIONAL SUBROUTINES-- 64
C -PRTCN4- USED TO PRODUCE PRINTER CONTOUR MAPS OF THE COMBINED 66
C UTILITY SURFACES. (IF MAPPING OPTION REQUESTED.) 68
C 70
C ----- 72
C 74
C DESCRIPTION OF CONTROL CARDS -- 76
C 78
C FIRST CONTROL CARD -- 80
C 82
C 1X-(COLS 1-3) - COLUMN OF DESIRED ORIGIN NODE 84
C 1Y-(COLS 4-6) - ROW OF DESIRED ORIGIN NODE 86
C 1X-(COLS 7-9) - COLUMN OF DESIRED DESTINATION NODE 88
C 1Y-(COLS 10-12) - ROW OF DESIRED DESTINATION NODE 90
C NFAC-(COLS 13-14) - NUMBER OF FACTORS TO BE INCLUDED IN THIS 92
C ANALYSIS 94
C NCNCE-(COLS 15-16) - NUMBER OF ALTERNATIVES TO BE GENERATED 96
C (MAXIMUM = 7, DEFAULT OPTION = 5) 98
C WIDTH-(COLS 17-20) - WIDTH OF PATH MAP AND CONTOUR MAP IN INCHES 100
C (PUNCH DECIMAL POINT, MAXIMUM VALUE 102
C ALLOWED = 12.7, DEFAULT OPTION = 10.0) 104
C MATRIX-(COLS 21-25) - THE WORD #TAPE3# INDICATES THE VALUE 106
C MATRICES ARE TO BE READ OFF A TAPE RATHER 108
C THAN FROM CARDS. THE TAPE SHOULD CONTAIN 110
C A STANDARD GCARS HEADER CARD, FOLLOWED BY 112
C THE MATRIX STORED ROW-WISE. 114
C (DEFAULT OPTION = 116
C IF THESE COLUMNS ARE BLANK, THE PROGRAM 118
C WILL EXPECT THE VALUE MATRICES AND THEIR 120
C HEADER CARDS TO BE SUPPLIED AS DATA CARDS. 122
C 124
C 1M-(COLS 26-35) - THE WORD #CONTOURS# CAUSES SUBROUTINE 126
C PRTCN4 TO BE CALLED TO PRODUCE A CONTOUR 128
C MAP OF THE FINAL COMBINED UTILITY SURFACE. 130
C (DEFAULT OPTION= NO MAP.) 132
C 1P-(COLS 36-40) - THE WORD #PUNCH# CAUSES A DECK OF CARDS TO 134
C BE PUNCHED CONTAINING THE FINAL COMBINED 136
C UTILITY SURFACE. 138
C (DEFAULT OPTION= NO CARDS.) 140
C 1F1X-(COLS 41-45) - THE WORD #DEHUG# CAUSES EXTRA PRINTED OUT- 142
C PUT TO BE PRODUCED. ADDITIONAL OUTPUT 144
C INCLUDES THE INPUT DATA (VALUE) MATRICES, 146
C THE UTILITY MATRICES, AND DETAILED LINK 148
C SUMMARIES OF EACH ALTERNATIVE. 150
C (DEFAULT OPTION= STANDARD OUTPUT AND PATH 152
C MAP ONLY.) 154
C 156

```

SECOND CONTROL CARD --

THIS CARD CONTAINS #NFACT# WEIGHTING VALUES. EACH COMPONENT UTILITY MATRIX WILL BE MULTIPLIED BY THE APPROPRIATE WEIGHT BEFORE ADDITION TO FORM THE FINAL COMBINED UTILITY MATRIX. THESE WEIGHTING FACTORS MUST THEREFORE BE IN THE CORRECT ORDER, SO THAT THE DESIRED FACTOR WILL BE MULTIPLIED BY THE DESIRED WEIGHT.

EACH WEIGHTING FACTOR IS ASSIGNED A FIVE COLUMN FIELD (1-5, 6-10, 11-15, ETC.) SO THAT UP TO 16 FACTORS CAN BE ENTERED ON A SINGLE CARD. IF MORE THAN 16 FACTORS ARE BEING COMBINED A SECOND (OR EVEN THIRD) CARD SHOULD BE USED. DECIMAL POINTS SHOULD BE PUNCHED. EACH FACTOR CAN THEN BE PLACED ANYWHERE WITHIN THE CORRECT 5 COLUMN FIELD. (DEFAULT OPTION- NONE)

THE FOLLOWING CARDS ARE REQUIRED ONLY IF A CONTOUR MAP HAS BEEN REQUESTED.-

MAP PARAMETER CARD--

CON-(COLS 1-2) - THE NUMBER OF CONTOUR INTERVALS (MAXIMUM = 19). (CON = DATA RANGE / DESIRED CONTOUR INTERVAL).

TOUR-(COL 3) - IF TOUR = 0, PROGRAM USES MAXIMUM AND MINIMUM FROM DATA FOR UPPER AND LOWER CONTOUR LIMITS.
IF TOUR = 1, DESIRED TOP AND BOTTOM CONTOURS READ IN FROM SPECIAL PROGRAM CARD (SEE BELOW).
IF TOUR = 2, VARIABLE CONTOURS READ IN (LOW TO HIGH). IN THIS CASE SET CON EQUAL TO THE NUMBER OF VALUES TO BE READ IN.
IF TOUR = 3, CONTOUR INTERVALS BECOME 1/2 STANDARD DEVIATION (MAX = +3 SIGMA, MIN = -3 SIGMA).

LINES-(COL 4) - IF LINES = 0, CONTOUR LINES ARE PRINTED.
IF LINES = 1, ALTERNATE (EVEN VALUED) CONTOUR BANDS ARE PRINTED.
IF LINES = 2, ALL BANDS PRINTED (ALL PRINT POSITIONS).

NEXT CARD(S) REQUIRED ONLY IF TOUR = 1 OR 2.

IF TOUR=1, THIS CARD SPECIFIES MINIMUM AND MAXIMUM ELEVATION VALUES. -- ZMIN(COLS 1-10) AND ZMAX(COLS 11-20).
IF TOUR=2, THIS CARD SPECIFIES A FORMAT (INCON) TO BE USED IN READING IN A SERIES OF IRREGULARLY SPACED CONTOUR LINES. THE VALUES ARE ARRANGED FROM LOWEST TO HIGHEST ON SUBSEQUENT CARDS.

THE FOLLOWING DATA CARDS ARE OMITTED WHEN #TAPE3# INPUT DATA IS REQUESTED-

-OTHERWISE-

THERE SHOULD BE #NFACT# DATA SETS EACH CONTAINING

- 1) A HEADER CARD WITH-
 - FACTOR-(COLS 1-40)- ANY 40 ALPHANUMERIC CHARACTERS IDENTIFYING FACTOR.
 - AREA-(COLS 41-70) - ANY 30 ALPHANUMERIC CHARACTERS IDENTIFYING AREA.
 - NCOL-(COLS 71-73) - NUMBER OF COLUMNS IN VALUE MATRIX.
 - NRROW-(COLS 74-76) - NUMBER OF ROWS IN VALUE MATRIX.
 - ITYPE-(COLS 77-79)- IF ITYPE = 1, VALUE SURFACE IS TOPOGRAPHIC RESIDUALS.
IF ITYPE = 2, VALUE SURFACE MEASURED IN TERMS OF COSTS.
IF ITYPE = 3, VALUE SURFACE MEASURED IN TERMS OF BENEFITS.
- 2) MATRIX CARDS - CONTAINING VALUE MATRIX IN ROWS IN STANDARD FORMAT (MF10.3). EACH NEW ROW STARTS ON A FRESH CARD, AND IS STORED ON NCOL/5 CARDS.

162
164
166
168
170
172
174
176
178
180
182
184
186
188
190
192
194
196
198
200
202
204
206
208
210
212
214
216
218
220
222
224
226
228
230
232
234
236
238
240
242
244
246
248
250
252
254
256
258
260
262
264
266
268
270
272
274
276
278
280
282
284
286
288
290
292
294
296
298
300
302
304
306
308
310
312
314
316
318
320

```

2.  DIMENSION TOTAL(7), PATIO(7), NUMLK(7) 322
3.  DIMENSION Z(50,50), ZZ(50,50), VAL(16) 324
4.  DIMENSION FACTOR(4), AREA(3) 326
5.  DIMENSION ICODE(4) 328
6.  DIMENSION TITL(8) 330
7.  DIMENSION NXJ(9000), TLINK(9000), DUMMY(1614) 332
8.  DATA (TITL(I),I=1,5)/10HCONTOUR MAP,10HP OF THE C,10HONSTRUCTED,10H 334
9.  1 UTILITY 5,10HURFACE / 336
10. DATA (ICODE(I),I=1,4)/5HPUNCH,10HCONTOURS ,5HDERUG,5HTAPE3/ 338
11. DATA IPACK/40000H/ 340
12. COMMON NXJ,TLINK,Z,ZZ,AREA,FACTOR,NCOL,NROW,KOUNT,NUMLK,K,N,J,REAL 342
13. 1,IOUT,VALUE,DUMMY,NHOME,NDEST,NCHCE,IM,IP,IFIX,NFAC 344
14. COMMON /MAPS/ CON,10UR,LINES,WIDTH,1HYP50,INCON(R),URANG(25),ZMIN, 346
15. 1ZMAX 348
16. INTEGER CON,TOUR 350
17. INTEGER SYSIN,SDF3 352
18. DATA SYSIN,SDF3/5,3/ 354
19. REAL MAXZ,MINZ 356
20. NVAL(I,J)=(I-1)*NCOL+J 358
21. XDIST=1.0 360
22. YDIST=1.0 362
23. C ----- 364
24. C INPUT ROUTINE 366
25. C ----- 368
26. 5 READ (SYSIN,140) IX,IY,JX,JY,NFAC,NCHCE,WIDTH,MATRIX,IM,IP,IFIX 370
27. IF (FOR,SYSIN) 135,10 372
28. 10 READ (SYSIN,145) (VAL(I),I=1,NFAC) 374
29. IF (IM,NE,ICODE(2)) GO TO 25 376
30. C ----- 378
31. C READ CONTOUR MAP CONTROLS ONLY IF CONTOUR MAP REQUESTED 380
32. C ----- 382
33. 24. READ (SYSIN,150) CON,TOUP,LINES 384
34. 1HYP50=0 386
35. 25. IF (TOUR,GT,2.0H,TOUR,LE,0) GO TO 25 388
36. 27. GO TO (15,20), TOUR 390
37. 28. 15 READ (SYSIN,155) ZMIN,ZMAX 392
38. 29. GO TO 25 394
39. C ----- 396
40. 30. 20 READ (SYSIN,160) INCON 398
41. READ (SYSIN,INCON) (URANG(I),I=1,CON) 400
42. 31. 25 DO 30 I=1,50 402
43. DO 30 J=1,50 404
44. 32. 30 ZZ(I,J)=0.0 406
45. 33. DO 100 N=1,NFAC 408
46. 34. 30 VAL(N)=VAL(N) 410
47. 35. 36. C ----- 412
48. C READ VALUE MATRICES FROM CARDS OR TAPE ACCORDING TO CONTROL REQUEST 414
49. C ----- 416
50. 37. IF (MATRIX,EQ,ICODE(4)) GO TO 40 418
51. 38. READ (SYSIN,165) FACTOR,AREA,NCOL,NROW,ITYPE 420
52. 39. DO 35 I=1,NROW 422
53. 40. 35 READ (SYSIN,170) (Z(I,J),J=1,NCOL) 424
54. 41. GO TO 50 426
55. C ----- 428
56. 42. 40 READ (SDF3) FACTOR,AREA,NCOL,NROW,ITYPE 430
57. 43. DO 45 I=1,NROW 432
58. 44. 45 READ (SDF3) (Z(I,J),J=1,NCOL) 434
59. 45. 50 IF (N,EQ,1) CALL PRTOUT (1) 436
60. 46. CALL PRTOUT (2) 438
61. C ----- 440
62. C WRITE DEBUG PRINTOUTS ONLY WHEN REQUESTED 442
63. C ----- 444
64. 47. IF (IFIX,EQ,ICODE(3)) CALL PRTOUT (9) 446
65. 48. GO TO (55,70,80), ITYPE 448
66. C ----- 450
67. C CONVERT VALUES TO UTILITIES SO AS TO MINIMIZE RESIDUALS 452
68. C ----- 454
69. 49. 55 DO 60 I=1,NROW 456
70. 50. DO 60 J=1,NCOL 458
71. 51. 60 Z(I,J)=ABS(Z(I,J)) 460
72. 52. CALL STATS (NROW,NCOL,50,50,Z,ZHAR,XP,ZSIG,MAXZ,MINZ) 462
73. 53. DO 65 I=1,NROW 464
74. 54. DO 65 J=1,NCOL 466
75. 55. 65 Z(I,J)=(Z(I,J)-MINZ)/(MAXZ-MINZ)*10.0 468
76. 56. IF (IFIX,EQ,ICODE(3)) CALL PRTOUT (10) 470
77. 57. GO TO 40 472
78. C ----- 474

```

```

C -----
C CONVERT VALUES TO UTILITIES SO AS TO MINIMIZE COSTS
C -----
58. 70 CALL STATS (NROW,NCOL,50,50,Z,ZBAR,XP,ZSIG,MAXZ,MINZ)
59.    DO 75 I=1,NROW
60.    DO 75 J=1,NCOL
61.    Z(I,J)=((Z(I,J)-MINZ)/(MAXZ-MINZ))*10.0
62.    IF (IFIX.EQ.ICODE(3)) CALL PRTOUT (10)
63.    GO TO 90
C -----
C CONVERT VALUES TO UTILITIES SO AS TO MAXIMIZE BENEFITS
C -----
64. 80 CALL STATS (NROW,NCOL,50,50,Z,ZBAR,XP,ZSIG,MAXZ,MINZ)
65.    DO 85 I=1,NROW
66.    DO 85 J=1,NCOL
67.    H5 Z(I,J)=10.0-((Z(I,J)-MINZ)/(MAXZ-MINZ))*10.0
68.    IF (IFIX.EQ.ICODE(3)) CALL PRTOUT (10)
C -----
C SUM THE COMPONENT UTILITY SURFACE*WEIGHTS TO OBTAIN FINAL SURFACE
C -----
69. 90 DO 95 I=1,NROW
70.    DO 95 J=1,NCOL
71.    95 ZZ(I,J)=Z(I,J)*VAL(N)+ZZ(I,J)
72. 100 CONTINUE
73.    NHOME=(IY-1)*NCOL+IX
74.    NUFST=(JY-1)*NCOL+JX
C -----
C END OF INPUT ROUTINE
C -----
C THIS AREA INCLUDES CALLS TO PRTOUT TO LIST FINAL COMBINED UTILITY
C SURFACE, TO PUNCH OUT MATRIX IF REQUESTED,AND TO MAKE A MAP OF THESE
C -----
75. IF (IFIX.EQ.ICODE(3)) CALL PRTOUT (3)
76. IF (IP.EQ.ICODE(1)) CALL PRTOUT (12)
77. IF (IM.NE.ICODE(2)) GO TO 105
78. CALL PRTCON4 (ZZ,11L,NROW,NCOL,50,50)
79. 105 DO 110 I=1,NROW
80.    DO 110 J=1,NCOL
81.    110 Z(I,J)=ZZ(I,J)
C -----
C GENERATE THE LINK TABLE
C -----
82. NUM=1
83. DO 130 I=1,NROW
84.    IUP=I-1
85.    IDN=I+1
86.    DO 130 J=1,NCOL
87.    JLFT=J-1
88.    JRT=J+1
89.    IF (JLFT.LT.1) GO TO 115
90.    NXJ(NUM)=NVAL(I,J)*IPACK+NVAL(I,JLFT)
91.    TLINK(NUM)=(Z(I,JLFT)+Z(I,J))*XU1ST/2.0
92.    NUM=NUM+1
93.    IF (JRT.GT.NCOL) GO TO 120
94.    NXJ(NUM)=NVAL(I,J)*IPACK+NVAL(I,JRT)
95.    TLINK(NUM)=(Z(I,JRT)+Z(I,J))*XD1ST/2.0
96.    NUM=NUM+1
97.    IF (IUP.LT.1) GO TO 125
98.    NXJ(NUM)=NVAL(I,J)*IPACK+NVAL(IUP,J)
99.    TLINK(NUM)=(Z(IUP,J)+Z(I,J))*XU1ST/2.0
100.    NUM=NUM+1
101.    125 IF (IDN.GT.NROW) GO TO 130
102.    NXJ(NUM)=NVAL(I,J)*IPACK+NVAL(IDN,J)
103.    TLINK(NUM)=(Z(IDN,J)+Z(I,J))*XD1ST/2.0
104.    NUM=NUM+1
105. 130 CONTINUE
106.    NSUM=NUM-1
C -----
C CALL MINIMUM PATH ROUTINE TO GENERATE ALTERNATIVES
C -----
107. CALL MPATH (NSUM,WIDTH)
108. GO TO 5
C -----
109. 135 STOP

```

	C	----- F O R M A T S -----	
110.	140	FORMAT (4I3,2I2,F4.0,A5,A10,2A5)	628
111.	145	FORMAT (16F5.0)	630
112.	150	FORMAT (12,2I1)	632
113.	155	FORMAT (2F10.0)	634
114.	160	FORMAT (8A10)	636
115.	165	FORMAT (4A10,3A10,2I3,I2)	638
116.	170	FORMAT (8F10.3)	640
117.		END	642
			644
			646-

c) FORSTAT CROSS-REFERENCE LISTINGS FOR PROGRAM NETSUM

PROGRAM NETSUM

STATEMENT NUMBER	DEFINITION	REFERENCES
5	20.	108.
10	22.	21.
15	26.	27.
20	30.	27.
25	32.	23. 26. 29.
30	34.	32. 33.
35	40.	39.
40	42.	37.
45	44.	43.
50	45.	41.
55	49.	48.
60	51.	49. 50.
65	55.	53. 54.
70	58.	48.
75	61.	59. 60.
80	64.	48.
85	67.	65. 66.
90	69.	57. 63.
95	71.	69. 70.
100	72.	35.
105	79.	77.
110	81.	79. 80.
115	93.	89.
120	97.	93.
125	101.	97.
130	105.	83. 86. 101.
135	109.	21.
140	110. FORMAT	20.
145	111. FORMAT	22.
150	112. FORMAT	24.
155	113. FORMAT	28.
160	114. FORMAT	30.
165	115. FORMAT	38.
170	116. FORMAT	40.

14 STATEMENT NUMBERS DEFINED

5 MAXIMUM NUMBER OF REFERENCES

46 TOTAL NUMBER OF REFERENCES

1.4 AVERAGE NUMBER OF REFERENCES

NIM	I	H2	91.	92.	94.	95.	96.	98.	99.	100.	102.	103.	104.	106.
NUMK	I	2. CD	90.											
NVAL	I	17. ASF	11.	98.	102.									
NXJ	I	7. CD	90.	94.	98.	102.								
PRICON4	P	78.	11.	90.	94.									
PROUT	P	48.						76.						
RATIO	R	2. D	46.	56.	62.	68.	75.							
RVAL	R	11. C												
SOF3	I	14.	15.	44.										
STATS	P	52.	58.											
SYSTN	I	14.	15.	21.	22.	24.	28.	30.	31.	38.	40.			
TITL	P	6. D	8.	78.										
TLINK	P	7. CD	11.	91.	95.	99.	103.							
TOTAL	P	2. D												
TOUR	I	12. C	13.	24.	26.	27.								
VAL	P	3. D	22.	36.	71.									
VALUF	P	11. C	36.											
WTOH	P	12. C	20.	107.										
XOIST	P	18.	91.	95.	99.	103.								
XP	P	52.	58.	64.										
YOIST	P	15.												
Z	P	3. CD	11.	40.	44.	51.	52.	55.	58.	61.	64.	67.	71.	81.
ZAAP	P		99.	103.										95.
ZMAX	P	52.	58.	64.										
ZMIN	P	12. C	28.											
ZSIG	P	12. C	28.											
ZZ	P	52.	58.	64.										
	P	3. CD	11.	34.	71.	78.	81.							

49 SYM-TOLS DEFINED
33 MAXIMUM NUMBER OF REFERENCES
273 TOTAL NUMBER OF REFERENCES
4.6 AVERAGE NUMBER OF REFERENCES

d) LISTING OF SUBROUTINE MPATH

```

1. SUBROUTINE MPATH (NLINK,WIDTH)
C SUBROUTINE MPATH
C-----
C PURPOSE --
C
C   TO GENERATE A SERIES OF ALTERNATIVES BY REPEATED APPLICATION OF MIN
C   MINIMUM PATH ANALYSIS TO A UTILITY NETWORK DEVELOPED BY PROGRAM MIN
C   NETSUM.
C   THIS SUBROUTINE USES A MODIFIED VERSION OF THE BRITISH ROAD MIN
C   RESEARCH LABORATORY (RRL) ALGORITHM. THE BASIS OF THIS FORTRAN COD-MIN
C   ING WAS DEVELOPED BY -
C   B. V. MARTIN, 1963,
C   MINIMUM PATH ALGORITHMS FOR TRANSPORTATION PLANNING MIN
C   CIVIL ENGINEERING SYSTEMS LABORATORY RESEARCH MIN
C   REPORT R63-52,
C   MASSACHUSETTS INSTITUTE OF TECHNOLOGY, MIN
C   CAMBRIDGE, MASS. MIN
C-----
C   THE GCARS SYSTEM
C   PURDUE UNIVERSITY
C
C   A. KEITH TURNER,
C   CIVIL ENGINEERING,
C   JUNE 1969.
C-----
C USAGE --
C
C   CALL MPATH (NLINK,WIDTH)
C   WHERE - NLINK = NUMBER OF LINKS IN NETWORK
C   - WIDTH = WIDTH OF MAP OF PATHS IN INCHES (TO BE PRO-
C   DUCED BY SUBROUTINE PATHS)
C-----
C DESCRIPTION OF CONTROL CARDS --
C
C   NONE REQUIRED- ALL NECESSARY VALUES SUPPLIED BY PROGRAM NETSUM MIN
C VIA COMMON STATEMENT MIN
C-----
C REQUIRED SUBROUTINES-
C   -PATHS
C   -PRTOUT
C
C OPTIONAL SUBROUTINES-
C   -NONE
C-----
2. DIMENSION AREA(3), FACTOR(4), NXJ(9800), TLINK(9800), NN(200) MIN
3. DIMENSION TSUM(2500), TCUM(1000), NCUM(1000), ITAB(200), NODE(200, MIN
17) MIN
4. DIMENSION TOTAL(7), RATIO(7), NUMLK(7), ICODE(3) MIN
5. DATA IPACK/400000/, (ICODE(I), I=1,3)/5HPUNCH,5HMAP ,5HDEBUG/ MIN
6. COMMON NXJ,TLINK,NN,TSUM,AREA,FACTOR,NCOL,NROW,KOUNT,NUMLK,K,N,J,RMIN
1VAL,IOUT,VALUE,TOTAL,RATIO,ITAB,NODE,NHOME,NDEST,NCHCE,I,M,IP,IFIX, MIN
2NFAC MIN
7. DATA (ICODE(I), I=1,3)/5HPUNCH,5HMAP ,5HDEBUG/ MIN
C INITIALIZE ALL ARRAYS
C-----
8. KOUNT=0
9. NZONE=NCOL*NROW
10. LNKDE=NLINK/NZONE
11. IF (NCHCE.LT.1.0R.NCHCE.GT.7) NCHCE=5
12. CALL PRTOUT (11)
13. 5 DO 10 I=1,NZONE
14. 10 TSUM(I)=99999.99
15. NN(NHOME)=0
16. TSUM(NHOME)=0
17. NCM=0
18. P=NNHOME
19. NCM=NNHOME+LNKDE

```

	C	-----	MIN	158
	C	SEARCH LINK TABLE FOR LINKS FROM NHOME.	MIN	160
	C	START SEARCH AT ENTRY NMM.	MIN	162
	C	-----	MIN	164
20.	15	IF ((NXJ(NMM)/IPACK)-NM) 25,20,20	MIN	166
21.	20	NMM=NMM-(LNKNDE*1)	MIN	168
22.		GO TO 15	MIN	170
	C	-----	MIN	172
23.	25	DO 40 I=NMM,NLINK	MIN	174
24.		N=NXJ(I)/IPACK	MIN	176
25.		IF (N=NM) 40,30,45	MIN	178
26.	30	J=NXJ(I)-N*IPACK	MIN	180
27.		K=J	MIN	182
	C	-----	MIN	184
	C	IF TSUM OF END NODE EQUALS 99999,99,	MIN	186
	C	ENTER LINK IN CUMULATIVE TABLE	MIN	188
	C	-----	MIN	190
28.		IF (TSUM(K)-99999,99) 40,35,40	MIN	192
29.	35	NCM=NCM+1	MIN	194
30.		TCUM(NCM)=TSUM(NM)+TLINK(I)	MIN	196
31.		NCUM(NCM)=J	MIN	198
32.	40	CONTINUE	MIN	200
	C	-----	MIN	202
	C	FIND MINIMUM TCUM IN CUMULATIVE TABLE	MIN	204
	C	-----	MIN	206
33.	45	TMIN=99999,99	MIN	208
34.		DO 55 K=1,NCM	MIN	210
35.		IF (TMIN-TCUM(K)) 55,55,50	MIN	212
36.	50	TMIN=TCUM(K)	MIN	214
37.		L=NCUM(K)	MIN	216
38.		M=K	MIN	218
39.	55	CONTINUE	MIN	220
40.		N=NXJ(L)/IPACK	MIN	222
41.		K=NXJ(L)-N*IPACK	MIN	224
42.		IF (TSUM(K)-TMIN) 60,60,65	MIN	226
43.	60	I=I	MIN	228
44.		GO TO 70	MIN	230
	C	-----	MIN	232
	C	-----	MIN	234
	C	ENTER THIS MINIMUM LINK IN TREE TABLE	MIN	236
	C	-----	MIN	238
45.	65	TSUM(K)=TMIN	MIN	240
46.		NN(K)=L	MIN	242
47.		I=0	MIN	244
	C	-----	MIN	246
	C	IF END NODE EQUALS DESTINATION, PATH FOUND, GO TO 90.	MIN	248
	C	-----	MIN	250
48.		IF (NDEST-K) 70,90,70	MIN	252
	C	-----	MIN	254
	C	UPDATE CUMULATIVE TABLE ENTRIES.	MIN	256
	C	MAKE LAST END NODE NHOME AND RETURN TO 15.	MIN	258
	C	-----	MIN	260
49.	70	DO 75 NM=M,NCM	MIN	262
50.		TCUM(NM)=TCUM(NM+1)	MIN	264
51.		NCUM(NM)=NCUM(NM+1)	MIN	266
52.		IF (NM+1=NCM) 75,80,75	MIN	268
53.	75	CONTINUE	MIN	270
54.	80	NCM=NCM+1	MIN	272
55.		IF (I) 45,85,45	MIN	274
56.	85	NMM=K*LNKNDE	MIN	276
57.		NM=K	MIN	278
58.		GO TO 15	MIN	280
	C	-----	MIN	282
	C	-----	MIN	284
	C	DESTINATION REACHED, TRACE PATH FROM TREE TABLE, STORE PATH STATIS-	MIN	286
	C	ICS.	MIN	288
	C	-----	MIN	290
59.	90	KOUNT=KOUNT+1	MIN	292
60.		IOUT=0	MIN	294
61.		IF (IFIX.EQ.ICODE(3)) CALL PRTOUT (5)	MIN	296
62.	95	IF (IFIX.EQ.ICODE(3)) CALL PRTOUT (6)	MIN	298
63.		IOUT=IOUT+1	MIN	300
64.		NODE(IOUT,KOUNT)=K	MIN	302
65.		IF (K=NHOME) 100,105,100	MIN	304
66.	100	L=NN(K)	MIN	306
67.		ITAH(IOUT)=L	MIN	308
68.		K=NXJ(L)/IPACK	MIN	310
69.		GO TO 95	MIN	312

70.	C	105	CONTINUE	MIN 314
71.			IOUT=IOUT-1	MIN 316
72.			TOTAL(KOUNT)=ISUM(NDEST)	MIN 318
73.			RATIO(KOUNT)=TOTAL(KOUNT)/TOTAL(1)	MIN 320
74.			RVAL=RATIO(KOUNT)	MIN 322
75.			NUMLK(KOUNT)=IOUT	MIN 324
76.			IF (FIX.EQ.ICODE(3)) CALL PRTOU (7)	MIN 326
77.			IF (RATIO(KOUNT).GE.2.0.OR.KOUNT.GE.NCHCE) GO TO 115	MIN 328
	C		-----	MIN 330
	C		CALCULATE THE OVERLAPS	MIN 332
	C		-----	MIN 334
78.			INUM=IOUT*0.035+0.5	MIN 336
79.			IF (INUM.LT.1) INUM=1	MIN 338
80.			KNUM=IOUT-2*INUM	MIN 340
	C		-----	MIN 342
	C		RESET CENTRAL TLINK VALUES TO VERY LARGE NUMBERS	MIN 344
	C		-----	MIN 346
81.			DO 110 LL=1,KNUM	MIN 348
82.			LVAL=INUM+LL	MIN 350
83.			L=ITAB(LVAL)	MIN 352
84.			TLINK(L)=5000.0	MIN 354
85.			IF (FIX.EQ.ICODE(3)) CALL PRTOU (8)	MIN 356
86.			GO TO 5	MIN 358
	C		-----	MIN 360
	C		ALL CHOICES REQUESTED HAVE BEEN FOUND.	MIN 362
	C		CALL PRTOU TO DISPLAY PATH STATISTICS AND PATHS TO PREPARE A MAP.	MIN 364
	C		-----	MIN 366
87.			115 CALL PRTOU (4)	MIN 368
88.			CALL PATHS (WIDTH)	MIN 370
89.			RETURN	MIN 372
90.			END	MIN 374
				MIN 376
				MIN 378

e) FORSTAT CROSS-REFERENCE LISTINGS FOR SUBROUTINE MPATH

SUBROUTINE MPATH

STATEMENT NUMBER	DEFINITION	REFERENCES
5	13.	86.
10	14.	13.
15	20.	22. 58.
20	21.	20.
25	23.	20.
30	26.	25.
35	29.	28.
40	32.	23. 25. 28.
45	33.	25. 55.
50	36.	35.
55	39.	34. 35.
60	43.	42.
65	45.	42.
70	49.	44. 48.
75	53.	49. 52.
80	54.	52.
85	56.	55.
90	59.	48.
95	62.	69.
100	66.	65.
105	70.	65.
110	84.	81.
115	87.	77.

23 STATEMENT NUMBERS DEFINED

3 MAXIMUM NUMBER OF REFERENCES

30 TOTAL NUMBER OF REFERENCES

1.3 AVERAGE NUMBER OF REFERENCES

SUBROUTINE XPATH

LABEL	TYPE	DEFINITION	REFERENCES												
			6.	7.	13.	14.	23.	24.	26.	30.	31.	43.	47.	55.	
AREA	R	2, CD	6.												
FACTOR	R	2, CD	6.												
I	I	5.	7.												
ICODE	I	4, D	5.												
IFIX	I	4, C	61.												
I*	I	4, C													
INUM	I	7R.	79.												
IOUT	I	6, C	60.												
IP	I	6, C													
IPACK	I	5.	20.												
ITAB	I	3, CD	6.												
J	I	6, C	26.												
K	I	4, C	27.												
			28.												
			64.												
			81.												
KNUM	I	8, C													
KOUNT	I	6, C	8.												
L	I	37.	40.												
LL	I	81.	82.												
LNKNDE	I	10.	19.												
LVAL	I	82.	83.												
M	I	38.	49.												
N	I	6, C	24.												
NCHCE	I	6, C	11.												
NCM	I	17.	29.												
NCOL	I	4, C	9.												
NCUM	I	3, D	31.												
NDEST	I	6, C	37.												
NFAC	I	4, C	48.												
NNHOMF	I	4, C	15.												
NNLINK	I	1.	10.												
N*	I	1R.	20.												
NNM	I	19.	20.												
NNN	I	2, CD	6.												
NODE	I	3, CD	6.												
NROW	I	4, C	9.												
NUMLK	I	4, CD	6.												
NXJ	I	2, CD	10.												
NZONE	I	9.													
PATHS	R	8R.													
PRATOUT	R	12.	61.												
RATIO	R	4, CD	6.												
RVAL	R	6, C	74.												
TCUM	R	3, D	30.												
TLINK	R	2, CD	6.												
TWIN	R	33.	35.												
TOTAL	R	4, CD	6.												
TSUM	R	3, CD	6.												
VALUF	P	4, C													
WIDTH	P	1.													

4M SYMBOLS DEFINED

1R MAXIMUM NUMBER OF REFERENCES

177 TOTAL NUMBER OF REFERENCES

3.7 AVERAGE NUMBER OF REFERENCES

6.	7.	13.	14.	23.	24.	26.	30.	31.	43.	47.	55.
61.	62.	76.	76.	85.	62.	76.	85.				
80.	82.										
63.	64.										
67.	71.										
75.	78.										
26.	40.										
27.	83.										
28.	34.										
35.	36.										
66.	68.										
64.	65.										
59.	64.										
41.	46.										
21.	56.										
8.	59.										
40.	41.										
19.	21.										
49.	25.										
24.	26.										
11.	77.										
29.	30.										
31.	37.										
48.	72.										
15.	16.										
10.	23.										
20.	25.										
20.	21.										
6.	15.										
6.	64.										
9.	75.										
6.	20.										
10.	13.										
24.	26.										
76.	85.										
74.	77.										
35.	36.										
30.	84.										
36.	42.										
72.	73.										
14.	16.										
30.	42.										
45.	72.										
28.	30.										
45.	45.										
72.	72.										
45.	45.										
72.	72.										

4M SYMBOLS DEFINED

1R MAXIMUM NUMBER OF REFERENCES

177 TOTAL NUMBER OF REFERENCES

3.7 AVERAGE NUMBER OF REFERENCES

H6.

f) LISTING OF SUBROUTINE PRTOU

```

1. SUBROUTINE PRTOU (IKOUNT)                                OUT  2
   C SUBROUTINE PRTOU                                       OUT  4
   C -----OUT  6
   C OUT  8
   C PURPOSE - TO PRODUCE PRINTER OUTPUT FOR PROGRAM NETSUM AND SUB- OUT 10
   C ROUTINE MPATH.                                         OUT 12
   C -----OUT 14
   C OUT 16
   C THE GCARS SYSTEM                                       OUT 18
   C PURDUE UNIVERSITY                                     OUT 20
   C OUT 22
   C A. KEITH TURNER,                                       OUT 24
   C CIVIL ENGINEERING,                                     OUT 26
   C JUNE 1969,                                             OUT 28
   C OUT 30
   C -----OUT 32
   C OUT 34
   C USAGE-- CALL PRTOU(K)                                   OUT 36
   C WHERE- K = ARGUMENT USED IN COMPUTED GO TO STATEMENT IN ORDER OUT 38
   C TO SELECT CORRECT OUTPUT FOR EACH CALL.               OUT 40
   C -----OUT 42
   C OUT 44
   C DESCRIPTION OF CONTROL CARDS--                          OUT 46
   C NONE REQUIRED                                           OUT 48
   C -----OUT 50
   C OUT 52
   C -----OUT 54
   C OUT 56
   C REQUIRED SUBROUTINES--                                   OUT 58
   C NONE                                                   OUT 60
   C OPTIONAL SUBROUTINES-- \                               OUT 62
   C NONE                                                   OUT 64
   C -----OUT 66
   C OUT 68
   C -----OUT 70
2. INTEGER SYSOT                                           OUT 72
3. DIMENSION ITAB(200), NODE(200,7)                        OUT 74
4. DIMENSION AREA(3), FACTOR(4), NXJ(9800), TLINK(9800), Z(50,50) OUT 76
5. DIMENSION ZZ(50,50), TSUM(2500), NN(2500)              OUT 78
6. DIMENSION TOTAL(7), RATIO(7), NUMLK(7), ICODE(3)       OUT 80
7. EQUIVALENCE (Z,NN), (ZZ,TSUM)                          OUT 82
8. COMMON NXJ,TLINK,Z,ZZ,AREA,FACTOR,NCOL,NROW,KOUNT,NUMLK,K,N,J,RVALOUT 84
   C I,IOUT,VALUE,TOTAL,RATIO,ITAB,NODE,NHOME,NUEST,NCHCE,IM,IP,IFIX,NFAOUT 86
   C 2C                                                    OUT 88
9. DATA (ICODE(I),I=1,3)/5HPUNCH,5HMAP ,5HDEBBUG/        OUT 90
10. DATA IPACK/40000H/,SYSOT/6/                          OUT 92
   C -----OUT 94
   C OUTPUT ROUTINE - OUTPUT REQUEST DETERMINED BY IKOUNT OUT 96
   C -----OUT 98
11. GO TO (5,10,15,25,35,40,50,60,70,80,90,95), IKOUNT OUT 100
   C -----OUT 102
12. WRITE (SYSOT,155)                                       OUT 104
13. WRITE (SYSOT,160)                                       OUT 106
14. WRITE (SYSOT,165)                                       OUT 108
15. WRITE (SYSOT,160)                                       OUT 110
16. WRITE (SYSOT,170) AREA                                  OUT 112
17. WRITE (SYSOT,175)                                       OUT 114
18. WRITE (SYSOT,180)                                       OUT 116
19. WRITE (SYSOT,175)                                       OUT 118
20. GO TO 105                                               OUT 120
   C -----OUT 122
   C OUT 124
21. WRITE (SYSOT,185) VALUE,FACTOR                         OUT 126
22. WRITE (SYSOT,175)                                       OUT 128
23. IF (IP,EQ,ICODE(1)) PUNCH 230, N,NFAC,VALUE,FACTOR OUT 130
24. GO TO 105                                               OUT 132
   C -----OUT 134
   C OUT 136
25. WRITE (SYSOT,190) NCOL,NROW                             OUT 138
26. DO 20 I=1,NROW                                         OUT 140
27. WRITE (SYSOT,195)                                       OUT 142
28. WRITE (SYSOT,200) (ZZ(I,J),J=1,NCOL)                  OUT 144
29. GO TO 105                                               OUT 146
   C -----OUT 148
   C OUT 150

```

30.	25	WRITE (SYSOT,205)	OUT 152
31.		WRITE (SYSOT,175)	OUT 154
32.		WRITE (SYSOT,210)	OUT 156
33.		WRITE (SYSOT,175)	OUT 158
34.		DO 30 I=1,KOUNT	OUT 160
35.		WRITE (SYSOT,215) I,TOTAL(I),RATIO(I),NUMLK(I)	OUT 162
36.		WRITE (SYSOT,175)	OUT 164
37.	30	CONTINUE	OUT 166
38.		GO TO 105	OUT 168
	C		OUT 170
	C	-----	OUT 172
39.	35	WRITE (SYSOT,220) KOUNT	OUT 174
40.		WRITE (SYSOT,225)	OUT 176
41.		GO TO 105	OUT 178
	C		OUT 180
	C	-----	OUT 182
42.	40	X1=(FLOAT(K)/FLOAT(NCOL))*1.0	OUT 184
43.		IX1=X1	OUT 186
44.		IY1=((X1-IX1)*FLOAT(NCOL))+0.3	OUT 188
45.		IF (IY1,NE,n) GO TO 45	OUT 190
46.		IX1=IX1-1	OUT 192
47.		IY1=NCOL	OUT 194
48.	45	WRITE (SYSOT,110) IX1,IY1,TSUM(K),K	OUT 196
49.		GO TO 105	OUT 198
	C		OUT 200
	C	-----	OUT 202
50.	50	WRITE (SYSOT,140) RVAL	OUT 204
51.		WRITE (SYSOT,115) IOUT	OUT 206
52.		WRITE (SYSOT,120)	OUT 208
53.		WRITE (SYSOT,130)	OUT 210
54.		DO 55 IV=1,IOUT	OUT 212
55.		L=ITAB(IV)	OUT 214
56.		N=NXJ(L)/IPACK	OUT 216
57.		J=NXJ(L)-N*IPACK	OUT 218
58.		WRITE (SYSOT,125) L,N,J,TLINK(L)	OUT 220
59.	55	CONTINUE	OUT 222
60.		GO TO 105	OUT 224
	C		OUT 226
	C	-----	OUT 228
61.	60	WRITE (SYSOT,135)	OUT 230
62.		WRITE (SYSOT,130)	OUT 232
63.		DO 65 IV=1,IOUT	OUT 234
64.		L=ITAB(IV)	OUT 236
65.		N=NXJ(L)/IPACK	OUT 238
66.		J=NXJ(L)-N*IPACK	OUT 240
67.		WRITE (SYSOT,125) L,N,J,TLINK(L)	OUT 242
68.	65	CONTINUE	OUT 244
69.		GO TO 105	OUT 246
	C		OUT 248
	C	-----	OUT 250
70.	70	WRITE (SYSOT,150) FACTOR	OUT 252
71.		DO 75 I=1,NROW	OUT 254
72.		WRITE (SYSOT,195)	OUT 256
73.		WRITE (SYSOT,200) (Z(I,J),J=1,NCOL)	OUT 258
74.	75	CONTINUE	OUT 260
75.		GO TO 105	OUT 262
	C		OUT 264
	C	-----	OUT 266
76.	80	WRITE (SYSOT,145) FACTOR	OUT 268
77.		DO 85 I=1,NROW	OUT 270
78.		WRITE (SYSOT,195)	OUT 272
79.		WRITE (SYSOT,200) (Z(I,J),J=1,NCOL)	OUT 274
80.	85	CONTINUE	OUT 276
81.		GO TO 105	OUT 278
	C		OUT 280
	C	-----	OUT 282
82.	90	WRITE (SYSOT,245)	OUT 284
83.		GO TO 105	OUT 286
	C		OUT 288
	C	-----	OUT 290
84.	95	PUNCH 235, NCOL,NROW	OUT 292
85.		DO 100 I=1,NROW	OUT 294
86.	100	PUNCH 240, (ZZ(I,J),J=1,NCOL)	OUT 296
87.	105	RETURN	OUT 298

```

C ----- F O R M A T S -----OUT 300
C                                     OUT 302
88. 110 FORMAT (1H ,5X,I2,1H,,I2,3X,F15,3,20X,I5)      OUT 304
89. 115 FORMAT (1H0,37HTOTAL NUMBER OF LINKS IN THIS PATH = ,I5)  OUT 306
90. 120 FORMAT (1H0,35HORIGINAL LINK VALUES FOR THIS PATH,)      OUT 308
91. 125 FORMAT (1H ,5X,I5,5X,I5,5X,I5,5X,F15,3)      OUT 310
92. 130 FORMAT (1H0,8X,1HL,10X,1HN,10X,1HJ,10X,5HTLINK,/)      OUT 312
93. 135 FORMAT (1H0,34HREVISED LINK VALUES FOR THIS PATH,)      OUT 314
94. 140 FORMAT (1H0,12HTHIS PATH IS,F10,3,23H TIMES THE FIRST CHOICE)  OUT 316
95. 145 FORMAT (1H0,26HSCALED UTILITY MATRIX FOR ,4A10)      OUT 318
96. 150 FORMAT (1H0,53HDEBBUG PRINTOUTS REQUESTED, LISTING OF INPUT DATA FOUT 320
    1R ,4A10,/)      OUT 322
97. 155 FORMAT (1H1)      OUT 324
98. 160 FORMAT (1H ,55X,29H-----)      OUT 326
99. 165 FORMAT (1H ,55X,29HI      THE GCARS SYSTEM      I,/,1H ,55X,29HI PUROUT 328
    1DUE UNIVERSITY VERSION I)      OUT 330
100. 170 FORMAT (///1H ,53HSEARCH ACTIVITY - PREPARATION OF THE UTILITY SUROUT 332
    1FACE,///,1H ,20X,20HTHIS ANALYSIS IS FOR,3A10,///)      OUT 334
101. 175 FORMAT (1H ,51H-----)      OUT 336
    1)      OUT 338
102. 180 FORMAT (2H 1,8X,1HI,40X,1HI,/,11H I WEIGHT I,12X,16HFACTORS UTILIZOUT 340
    1ED,12X,1HI,/,2H 1,8X,1HI,40X,1HI)      OUT 342
103. 185 FORMAT (2H 1,8X,1HI,40X,1HI,/,2H I,2X,F5,2,2H 1,4A10,1HI)      OUT 344
104. 190 FORMAT (1H1,23HTHIS UTILITY MATRIX OF ,I3,12H COLUMNS BY ,I3,46H ROUT 346
    1OWS REPRESENTS THE COMBINED UTILITY SURFACE,////)      OUT 348
105. 195 FORMAT (1H0)      OUT 350
106. 200 FORMAT (12F10,3)      OUT 352
107. 205 FORMAT (1H0,26HTHE GENERATED ALTERNATIVES,/)      OUT 354
108. 210 FORMAT (2H 1,8X,1HI,12X,1HI,15X,1HI,11X,1HI,/,52H I CHOICE I PATH TOUT 356
    1OTAL I RELATIVE PATH I NUMBER OF I,/,11H I NUMBER I,12X,29HI      VOUT 358
    2ALUE I LINKS I,/,2H 1,8X,1HI,12X,1HI,15X,1HI,11X,1HI)      OUT 360
109. 215 FORMAT (2H 1,8X,1HI,12X,1HI,15X,1HI,11X,1HI,/,5H I ,I2,5H I ,FOUT 362
    110,3,6H I ,F7,3,4X,1HI,4X,I3,4X,1HI,/,2H 1,8X,1HI,12X,1HI,15X,1OUT 364
    2HI,11X,1HI)      OUT 366
110. 220 FORMAT (1H1,27HANALYSIS FOR CHOICE NUMBER ,I3)      OUT 368
111. 225 FORMAT (1H0,5X,22HMINIMUM PATH REQUESTED,///,5X,4HN00E,10X,8HPATH  OUT 370
    1SUM)      OUT 372
112. 230 FORMAT (7HNUMBER ,I2,4H OF ,I2,16H FACTORS,WEIGHT=,F5,2,4X,4A10)  OUT 374
113. 235 FORMAT (27H COMBINED UTILITY SURFACE,,43X,213,2H 2)      OUT 376
114. 240 FORMAT (8F10,3)      OUT 378
115. 245 FORMAT (1H1,45HSEARCH ACTIVITY - GENERATION OF ALTERNATIVES,////,OUT 380
    150H REPEATED USE OF A MODIFIED VERSION OF THE BRITISH,/,48H ROAD ROUT 382
    2ESEARCH LABORATORY MINIMUM PATH ALGORITHM,/,41H GENERATES AS MANY OUT 384
    3AS SEVEN ALTERNATIVES,/)      OUT 386
116.      END      OUT 388-

```


SUBROUTINE PRTOUT

LABEL	TYPE	DEFINITION	REFERENCES									
AREA	R	4. CD	8.	16.								
FACTOR	R	4. CD	8.	21.	23.	70.	76.					
FLOAT	R	42.	44.									
I	I	9.	26.	28.	34.	35.	71.	73.	77.	79.	85.	86.
ICODE	I	6. D	9.	23.								
IFIX	I	8. C										
IKOUNT	I	1.	11.									
IM	I	8. C										
IOUT	I	8. C	51.	54.	63.							
IP	I	8. C	23.									
IPACK	I	10.	56.	57.	65.	66.						
ITAB	I	3. CD	8.	55.	64.							
IV	I	54.	55.	63.	64.							
IX1	I	43.	44.	46.	48.							
IY1	I	44.	45.	47.	48.							
J	I	8. C	28.	57.	58.	66.	67.	73.	79.	86.		
K	I	8. C	42.	48.								
KOUNT	I	8. C	34.	39.								
L	I	55.	56.	57.	58.	64.	65.	66.	67.			
N	I	8. C	23.	56.	57.	58.	65.	66.	67.			
NCHCE	I	8. C	25.	28.	42.	44.	47.	73.	79.	84.	86.	
NCOL	I	8. C										
NDEST	I	8. C	23.									
NFAC	I	8. C										
NHOM	I	8. C										
NN	I	5. DE	7.									
NODE	I	3. CD	8.									
NRQM	I	8. C	25.	26.	71.	77.	84.	85.				
NUMLK	I	6. CD	8.	35.								
NXJ	I	4. CD	8.	35.	57.	65.	66.					
RATIO	R	6. CD	8.	35.								
RVAL	R	8. C	50.									
SV50T	I	2.	10.	12.	13.	14.	15.	16.	17.	18.	19.	21.
			30.	31.	32.	33.	35.	36.	39.	40.	48.	50.
			61.	62.	67.	70.	72.	73.	76.	78.	79.	82.
			8.	58.	67.							
TLINK	R	4. CD	8.	35.								
TOTAL	R	6. DE	7.	48.								
TSUM	R	8. C	21.	23.								
VALUE	R	42.	43.	44.								
X1	R		7.	8.	73.	79.						
Z	R	4. CDE	7.	8.	28.	86.						
ZZ	R	5. CDE	7.	8.								

40 SYMBOLS DEFINED		
38	MAXIMUM	NUMBER OF REFERENCES
152	TOTAL	NUMBER OF REFERENCES
3.8	AVERAGE	NUMBER OF REFERENCES

h) LISTING OF SUBROUTINE PATHS

1.	SUBROUTINE PATHS (WIDTH)	MAP	2
	SUBROUTINE PATHS	MAP	4
	-----	MAP	6
	PURPOSE --	MAP	8
		MAP	10
		MAP	12
	TO DEVELOP AND PRINT A MAP SHOWING THE LOCATIONS OF THE ALTER-	MAP	14
	NATIVES GENERATED BY SUBROUTINE MPATH.	MAP	16
	EACH ALTERNATIVE IS DEFINED BY A STRING OF NUMBERS WHICH COR-	MAP	18
	RESPOND TO THE CHOICE NUMBER (I.E. FIRST CHOICE DEFINED BY STRING	MAP	20
	OF ONES, ETC.). WHERE TWO OR MORE CHOICES OVERLAP OR CROSS, THE	MAP	22
	CHARACTER BELONGING TO THE LOWEST CHOICE NUMBER IS THE ONLY ONE	MAP	24
	PRINTED.	MAP	26
		MAP	28
	-----	MAP	30
	THE GCARS SYSTEM	MAP	32
	PURDUE UNIVERSITY	MAP	34
		MAP	36
	A. KEITH TURNER,	MAP	38
	CIVIL ENGINEERING,	MAP	40
	JUNE 1969.	MAP	42
		MAP	44
	-----	MAP	46
	USAGE --	MAP	48
		MAP	50
	CALL PATHS (WIDTH)	MAP	52
	WHERE= WIDTH = WIDTH OF MAP IN INCHES.	MAP	54
		MAP	56
		MAP	58
	-----	MAP	60
	DESCRIPTION OF CONTROL CARDS--	MAP	62
		MAP	64
		MAP	66
	NONE REQUIRED= ALL NECESSARY VALUES SUPPLIED BY PROGRAM NETSUM	MAP	68
	OR SUBROUTINE MPATH VIA COMMON STATEMENT.	MAP	70
		MAP	72
	-----	MAP	74
	REQUIRED SUBROUTINES--	MAP	76
	-NONE	MAP	78
	OPTIONAL SUBROUTINES--	MAP	80
	-NONE	MAP	82
		MAP	84
		MAP	86
	-----	MAP	88
2.	DIMENSION AREA(3), FACTOR(4), SYM(129), PSYM(129), ISYMB(7)	MAP	90
3.	DIMENSION NUMLK(7)	MAP	92
4.	DIMENSION DUM1(24342), DUM2(220)	MAP	94
5.	DIMENSION NODE(200,7), ICODE(3)	MAP	96
6.	COMMON SYM,PSYM,DUM1,AREA,FACTOR,COLS,ROWS,KOUNT,NUMLK,DUM2,NODE,NMAP	MAP	98
	1HOME,NDEST,NCHCE	MAP	100
7.	INTEGER ROWS,COLS,COLMAX,COL,PSYM,PS,GRID,SYM,BLANK,PLUS,MINUS,XST	MAP	102
	1OP,YSTOP,EYE,SYSTOT	MAP	104
8.	DATA BLANK/1H /,PLUS/1H+/,MINUS/1H-/, (ISYMB(I),I=1,7)/1H1,1H2,1H3,MAP	MAP	106
	1H4,1H5,1H6,1H7/,EYE/1H1/,SYSTOT/6/	MAP	108
9.	DATA (ICODE(I),I=1,3)/SHPUNCH,SHMAP ,5HDEBUG/	MAP	110
10.	WRITE (SYSTOT,90) NCHCE,COLS,ROWS	MAP	112
	-----	MAP	114
	CALCULATE MAP WIDTH IN CHARACTERS (COL)	MAP	116
		MAP	118
	IF (WIDTH.LE.0.0) WIDTH=10.0	MAP	120
11.	IF (WIDTH.GT.12.7) WIDTH=12.7	MAP	122
12.	COL=WIDTH*10.0*2.0	MAP	124
13.	PS=(COL-COLS)/(COLS-1)	MAP	126
14.		MAP	128
	-----	MAP	130
	TEST GRID SIZE--IF GRID SMALLER THAN 2 CHARACTERS BETWEEN COLUMNS,	MAP	132
	AN ERROR STATEMENT IS PRINTED AND PLOT ABORTED.	MAP	134
		MAP	136
15.	IF (PS.GT.0) GO TO 5	MAP	138
16.	WRITE (SYSTOT,95)	MAP	140
17.	GO TO A5	MAP	142
		MAP	142

	C	-----	MAP 144
	C	CALCULATE THE REQUIRED CONSTANTS	MAP 146
	C	-----	MAP 148
18.	5	COLMAX=PS*(COLS-1)+COLS	MAP 150
19.		XSTOP=PS+1	MAP 152
20.		YSTOP=(PS+1)*6+5)/10	MAP 154
21.		NLAST=COLMAX+2	MAP 156
22.		NLAS=NLAST-1	MAP 158
23.		GRID=COLS-1	MAP 160
24.		IROW=ROWS+1	MAP 162
	C	-----	MAP 164
	C	SET UP COLUMN GRID MARKS	MAP 166
	C	-----	MAP 168
25.		COL=3	MAP 170
26.		DO 15 J=1,GRID	MAP 172
27.		DO 10 I=2,XSTOP	MAP 174
28.		SYM(COL)=BLANK	MAP 176
29.	10	COL=COL+1	MAP 178
30.		SYM(COL)=PLUS	MAP 180
31.	15	COL=COL+1	MAP 182
32.		SYM(1)=BLANK	MAP 184
33.		SYM(2)=PLUS	MAP 186
34.		SYM(NLAST)=BLANK	MAP 188
	C	-----	MAP 190
	C	PRINT MARKS ACROSS MAP TOP	MAP 192
	C	-----	MAP 194
35.		WRITE (SYSOT,100) (SYM(I),I=1,NLAST)	MAP 196
	C	-----	MAP 198
	C	ESTIMATE VALUES FOR ALL PRINTING POSITIONS ON THE MAP	MAP 200
	C	THIS IS DONE BY DEVELOPING THEN PRINTING ONE LINE AT A TIME	MAP 202
	C	-----	MAP 204
	C	LOOP THROUGH THE MAP LENGTH , MATRIX ROW BY MATRIX ROW	MAP 206
	C	-----	MAP 208
36.		ITOP=1	MAP 210
37.		DO 80 MM=2,IROW	MAP 212
38.		IF (MM,EQ,IROW) YSTOP=1	MAP 214
39.		MM=MM+1	MAP 216
	C	-----	MAP 218
	C	LOOP THROUGH GRID, LINE BY LINE.	MAP 220
	C	IF KJ = 1, LINE GOES THROUGH GRID NODE.	MAP 222
	C	IF KJ = 2, SECOND LINE IN GRID = FIRST ONE NOT THROUGH NODES.	MAP 224
	C	IF KJ = 3,4, ETC. ADDITIONAL LINES SAME AS JUST PREPARED.	MAP 226
	C	-----	MAP 228
40.		DO 80 KJ=1,YSTOP	MAP 230
41.		IF (KJ,GT,2) GO TO 80	MAP 232
42.		DO 20 N=1,129	MAP 234
43.	20	PSYM(N)=BLANK	MAP 236
44.		PSYM(2)=EYE	MAP 238
45.		PSYM(NLAS)=EYE	MAP 240
46.		IF (YSTOP,NE,1,AND,ITOP,NE,1) GO TO 30	MAP 242
	C	-----	MAP 244
	C	LOOP ACROSS MATRIX COLUMNS	MAP 246
	C	-----	MAP 248
47.		DO 25 N=2,NLAS	MAP 250
48.	25	PSYM(N)=MINUS	MAP 252
49.	30	COL=2	MAP 254
50.		IF (KJ,EQ,2) GO TO 55	MAP 256
	C	-----	MAP 258
	C	KJ = 1, A LINE THROUGH NODES	MAP 260
	C	-----	MAP 262
51.		PSYM(1)=PLUS	MAP 264
52.		PSYM(NLAST)=PLUS	MAP 266
53.		DO 50 J=1,GRID	MAP 268
54.		I2COL=COL	MAP 270
55.		I2COL=COL+XSTOP	MAP 272
56.		I2=(M-1)*COLS+J	MAP 274
57.		I2P=I2+1	MAP 276
58.		DO 45 N=1,NCHCE	MAP 278
59.		KK=NCHCE-N+1	MAP 280
60.		NLK=NUMLK(KK)+1	MAP 282
61.		ISW1=0	MAP 284
62.		ISW2=0	MAP 286

```

63.      DO 35 NN=1,NLK      MAP 288
64.          IF (IZ.EQ.NODE(NN,KK)) ISW1=1      MAP 290
65.      35  IF (IZP.EQ.NODE(NN,KK)) ISW2=1      MAP 292
66.          IF (ISW1.EQ.1) PSYM(IZCOL)=ISYMB(KK)      MAP 294
67.          IF (ISW2.EQ.1) PSYM(ICOL)=ISYMB(KK)      MAP 296
68.          IF (ISW1.EQ.0.OR.ISW2.EQ.0) GO TO 45      MAP 298
69.          COL=IZCOL      MAP 300
70.          DO 40 I=2,XSTOP      MAP 302
71.              COL=COL+1      MAP 304
72.          40  PSYM(COL)=ISYMB(KK)      MAP 306
73.      45  CONTINUE      MAP 308
74.      50  COL=ICOL      MAP 310
75.          GO TO 75      MAP 312
C      MAP 314
C      MAP 316
C      KJ = 2, FIRST INTERMEDIATE LINE      MAP 318
C      MAP 320
63.      55  DO 70 J=1,GRID      MAP 322
77.          ICOL=COL+XSTOP      MAP 324
78.          IZ=(M-1)*COLS+J      MAP 326
79.          IZP=IZ+1      MAP 328
80.          IZUL=IZ+COLS      MAP 330
81.          IZUR=IZP+COLS      MAP 332
82.          DO 65 N=1,NCHCE      MAP 334
83.              KK=NCHCE-N+1      MAP 336
84.              NLK=NUMLK(KK)+1      MAP 338
85.              ISW1=0      MAP 340
86.              ISW2=0      MAP 342
87.              ISW3=0      MAP 344
88.              ISW4=0      MAP 346
89.              DO 60 NN=1,NLK      MAP 348
90.                  IF (IZ.EQ.NODE(NN,KK)) ISW1=1      MAP 350
91.                  IF (IZP.EQ.NODE(NN,KK)) ISW2=1      MAP 352
92.                  IF (IZUL.EQ.NODE(NN,KK)) ISW3=1      MAP 354
93.      60  IF (IZUR.EQ.NODE(NN,KK)) ISW4=1      MAP 356
94.                  ITEST=ISW1      MAP 358
95.                  ITEST=ISW2      MAP 360
96.                  ITEST=ISW3      MAP 362
97.                  ITEST=ISW4      MAP 364
98.                  IF (ISW1.EQ.1.AND.ISW3.EQ.1) PSYM(COL)=ISYMB(KK)      MAP 366
99.                  IF (ISW2.EQ.1.AND.ISW4.EQ.1) PSYM(ICOL)=ISYMB(KK)      MAP 368
100.      65  CONTINUE      MAP 370
101.      70  COL=ICOL      MAP 372
102.      75  ITOP=0      MAP 374
C      MAP 376
C      PRINT OUT LINE JUST PREPARED      MAP 378
C      MAP 380
103.      80  WRITE (SYSOT,100) (PSYM(I),I=1,NLAST)      MAP 382
C      MAP 384
C      MAP FINISHED, PRINT MARKS ACROSS MAP BOTTOM      MAP 386
C      MAP 388
C      WRITE (SYSOT,100) (SYM(I),I=1,NLAST)      MAP 390
104.      85  RETURN      MAP 392
C      MAP 394
C      MAP 396
106.      90  FORMAT (1H0,/,31H MAP OF GENERATED ALTERNATIVES.,/,1H ,12,19H CHMAP 398
107.          10ICES ARE SHOWN.,/,45H EACH IS LOCATED BY A STRING OF NUMBERS WHICHMAP 400
108.          2H.,/33H CORRESPOND TO THE CHOICE NUMBER.,/,42H THE ALTERNATIVES AMAP 402
109.          3HE DEFINED ON A NETWORK,/,4H HAVING ,12,12H COLUMNS BY ,12,6H ROWSMAP 404
110.          4.,/))      MAP 406
107.      95  FORMAT (25H TOO MANY COLUMNS IN GRID)      MAP 408
108.      100  FORMAT (1H ,129A1)      MAP 410
109.          ENQ      MAP 412-

```

i) FORSTAT CROSS-REFERENCE LISTINGS FOR SUBROUTINE PATHS

SUBROUTINE PATHS

STATEMENT NUMBER	DEFINITION	REFERENCES
5	14.	15.
10	29.	27.
15	31.	26.
20	43.	42.
25	48.	47.
30	49.	46.
35	65.	63.
40	72.	70.
45	73.	58. 68.
50	74.	53.
55	76.	50.
60	93.	89.
65	100.	82.
70	101.	76.
75	102.	75.
80	103.	37. 40. 41.
85	105.	17.
90	106. FORMAT	10.
95	107. FORMAT	16.
100	108. FORMAT	35. 103. 104.

20 STATEMENT NUMBERS DEFINED

3 MAXIMUM NUMBER OF REFERENCES

25 TOTAL NUMBER OF REFERENCES

1.3 AVERAGE NUMBER OF REFERENCES

SUBROUTINE PATHS

LABEL	TYPE	DEFINITION	REFERENCES														
			6.	28.	32.	34.	43.										
AREA	R	2. CD	6.														
HLANK	I	7.	13.	14.	25.	28.	29.	30.	31.	49.	54.	55.	69.	71.	72.	74.	
COL	I	7.	77.	98.	101.												
COLMAX	I	7.	18.	21.													
COLS	I	6. C	7.	10.	14.	18.	23.	56.	78.	80.	81.						
DUM1	R	4. CD	6.														
DUM2	R	4. CD	6.														
EYE	I	7.	8.	44.	45.												
FACTOK	R	2. CD	23.	26.	53.	76.	103.	104.									
GR10	I	7.	9.	27.	35.	70.											
ICODE	I	5. D	9.														
ICOL	I	5.	67.	74.	77.	99.	101.										
IROW	I	24.	37.	38.	68.	85.	90.	94.	98.								
ISW1	I	61.	64.	66.	68.	86.	91.	95.	99.								
ISW2	I	62.	65.	67.	68.	86.	91.	95.	99.								
ISW3	I	87.	92.	96.	98.												
ISW4	I	AR.	93.	97.	99.												
ISYMH	I	2.	8.	66.	67.	72.	98.	99.									
ITEST	I	94.	95.	96.	97.												
ITOP	I	36.	49.	102.													
I7COL	I	56.	57.	64.	78.	79.	80.	90.									
I7COL	I	54.	66.	69.													
I7DL	I	80.	92.														
I7DR	I	81.	93.														
I7P	I	57.	65.	79.	81.	91.											
J	I	26.	53.	56.	76.	78.											
KJ	I	40.	41.	50.													
KK	I	59.	60.	64.	65.	66.	67.	72.	83.	84.	90.	91.	92.	93.	98.	99.	
KOUNT	I	4. C															
M	I	39.	56.	78.													
MINUS	I	37.	8.	48.													
MM	I	37.	38.	39.	48.	58.	59.	82.	83.								
N	I	42.	43.	47.	59.	82.	83.										
NCHCE	I	4. C															
NHCE	I	4. C															
NHOM	I	4. C															
NLAST	I	27.	45.	47.													
NLAS	I	21.	22.	34.	35.	52.	103.	104.									
NLK	I	60.	63.	84.	89.	90.	91.	92.	93.								
NN	I	63.	64.	65.	89.	90.	91.	92.	93.								
NONE	I	5. CD	6.	64.	65.	90.	91.	92.									
NUMK	I	3. CD	6.	60.	84.												
PLUS	I	7.	8.	30.	33.	51.	52.										
PS	I	7.	14.	15.	18.	19.	20.										
PSYM	I	2. CD	6.	7.	43.	44.	45.	48.	51.	52.	66.	67.	72.	98.	99.	103.	
ROWS	I	6. C	7.	10.	24.												
SYM	I	2. CD	6.	7.	28.	30.	32.	33.	34.	35.	103.	104.					
SYSTOT	I	7.	8.	10.	16.	35.	103.	104.									
WIDTH	R	1.	11.	12.	13.												
XSTOP	I	7.	19.	27.	55.	70.	77.										
YSTOP	I	7.	20.	38.	40.	46.											

52 SYMBOLS DELETED

17 MAXIMUM NUMBER OF REFERENCES

223 TOTAL NUMBER OF REFERENCES

4.3 AVERAGE NUMBER OF REFERENCES

j) EXAMPLE OF CONTROL CARDS DATA FOR PROGRAM NETSUM

2 12 24 32 2 512.7		CONTOURS PUNCH						
1.0	1.0							
1001								
EARTHWORK	COST FACTOR	NORTH			25 33 1			
13.368	13.130	3.071	-4.207	-10.072	-9.443	-11.249	-14.725	
-25.202	-20.409	-21.139	-9.460	-11.345	-8.477	2.83	10.574	
9.755	22.193	32.168	27.301	-50.099	-15.816	-40.023	-24.723	
33.917								
24.746	18.974	12.164	-4.743	-9.860	-5.301	-9.436	-19.145	
-18.306	-11.214	-11.509	-6.624	-2.695	-2.047	7.711	14.338	
33.669	32.947	38.330	33.321	24.661	23.658	-26.847	19.301	
-1.712								
26.445	31.967	30.121	-1.163	-7.961	-2.223	-6.799	-9.033	
-1.111	2.045	5.626	10.067	10.322	1.170	29.463	36.449	
42.922	30.004	34.715	30.211	-23.000	-36.051	-24.639	15.093	
-55.166								
15.450	18.932	5.056	-5.100	-7.570	-1.612	5.563	-1.035	
-256	10.751	9.715	19.624	19.459	17.355	39.304	33.649	
33.591	19.301	30.891	26.754	.051	5.128	-37.740	-58.459	
-64.740								
3.257	7.293	17.414	-6.547	5.165	-3.256	.137	5.170	
3.989	12.820	-2.392	9.004	14.054	13.240	37.194	35.095	
-3.101	3.734	28.883	12.076	6.078	-52.423	-57.055	-30.245	
21.597								
-8.275	-4.875	-10.491	.824	-19.419	20.608	7.701	-5.501	
7.077	15.218	23.548	39.537	28.989	21.397	-5.633	-8.135	
-10.090	10.669	-47.546	-26.037	-65.027	26.051	56.332	45.727	
23.392								
-21.775	-16.293	-8.888	-6.275	-15.369	-10.261	-6.147	3.427	
8.672	1.964	16.626	29.475	-11.181	-8.144	21.474	-27.340	
-38.731	-44.246	-28.144	-49.558	-61.141	-44.797	57.025	38.419	
-18.644								
-33.557	-29.221	-21.666	-25.050	-27.246	8.596	-9.098	4.247	
-5.103	-15.065	22.773	-2.190	-1.234	26.670	-22.639	-44.446	
-27.142	-27.987	-74.262	-48.969	34.499	49.938	53.873	40.954	
29.560								
-31.500	-28.825	-22.631	-16.594	3.555	1.553	-9.176	-15.752	
-11.154	12.331	20.842	33.719	27.998	-14.537	-48.740	-29.144	
-39.529	-65.788	.804	-27.176	29.709	40.404	15.786	29.246	
16.206								
-21.911	-17.129	-16.050	-11.453	-7.488	-4.406	-22.138	1.825	
26.143	43.319	40.795	48.324	-26.833	-76.765	-66.567	-74.803	
-69.303	-49.883	21.271	35.903	39.411	24.814	39.785	24.343	
21.203								
-14.791	-6.557	-11.900	-25.992	-27.877	-20.396	-.049	29.057	
36.630	48.915	53.870	34.531	-81.644	-78.096	-80.281	-23.078	
-50.929	24.740	26.065	41.812	48.005	43.640	39.989	35.131	
19.068								
-7.603	-6.231	-37.197	-33.856	-22.424	3.513	15.464	43.777	
52.459	61.180	108.396	-107.051	-92.599	-88.317	-86.421	-21.388	
8.555	16.241	25.127	48.155	45.561	41.342	34.812	26.919	
16.508								
-13.532	-7.573	-28.391	9.116	17.679	28.818	18.616	59.023	
74.861	85.030	80.496	-86.934	-89.743	-87.154	-75.454	-19.024	
2.789	16.263	33.786	47.208	42.163	23.849	17.004	18.070	
15.720								
-41.168	-.984	15.028	31.430	41.259	39.595	52.388	67.457	
89.543	92.729	-76.751	-80.728	-80.941	-61.163	-51.995	-47.172	
10.960	32.872	46.624	43.980	39.540	21.438	11.668	9.018	
-10.373								
29.094	34.071	38.649	36.165	33.241	57.728	60.397	83.024	
59.051	92.977	-78.409	-84.658	-73.262	-49.602	-25.225	-43.697	
38.672	41.080	31.207	19.246	7.287	-47.939	-52.139	-36.640	
-.502								

B-50

63.031	44.280	43.301	44.140	29.191	46.706	62.808	47.134
51.135	56.653	-73.695	-59.794	-24.427	2.264	33.642	2.644
-49.104	-10.012	-49.206	-70.941	-66.630	4.574	7.822	-41.474
8.327							
68.022	-8.240	58.094	48.646	33.470	46.705	18.091	7.352
2.874	-10.410	-87.266	-19.282	-3.526	20.741	56.041	66.231
62.541	-51.801	-71.833	13.196	9.787	-47.906	-47.862	8.317
10.732							
51.683	67.384	30.071	-9.180	28.835	-7.537	-4.403	-3.125
-12.929	-68.155	-51.204	-20.869	46.061	79.696	66.853	65.506
57.835	36.875	-27.555	11.628	8.433	-2.216	-41.397	39.144
-45.126							
-34.272	-59.298	-33.579	-47.004	-24.636	-15.515	-19.790	-19.963
-37.521	-80.202	-19.112	45.220	77.318	64.335	55.826	49.456
44.944	33.271	31.304	27.656	13.861	11.359	-42.843	-38.836
-10.715							
8.007	-60.214	-88.588	-94.789	-89.247	-98.118	-46.731	-58.828
-78.788	-86.934	38.552	46.296	49.976	45.612	50.371	48.852
38.394	29.572	18.098	13.011	20.941	15.035	-47.111	-17.832
-38.391							
38.107	70.548	-7.890	-19.402	-77.292	-47.768	-81.840	-78.302
-7.254	41.612	-30.372	16.634	36.517	40.399	35.899	29.948
19.684	15.661	16.555	21.627	27.115	20.569	-39.944	3.842
10.118							
40.924	30.614	31.406	49.002	-74.946	-57.783	-1.783	4.728
18.892	30.499	25.648	31.938	30.483	28.016	24.530	16.097
16.519	18.990	25.173	18.234	12.835	9.408	-15.977	-13.171
22.953							
-51.703	11.029	24.190	16.193	-2.539	-57.629	-27.753	15.221
18.876	32.169	31.319	27.213	23.725	11.900	15.611	9.777
4.516	4.273	1.770	5.224	12.425	13.361	9.865	-24.747
40.200							
-389.918	42.909	22.765	14.219	6.918	6.892	-50.856	-54.997
-18.599	8.345	17.402	22.379	8.199	3.953	7.715	1.192
9.097	1.918	3.368	2.104	6.030	8.426	12.984	-11.060
-19.482							
52.748	33.258	18.611	3.767	-1.319	.689	-10.819	-23.738
-43.441	-12.226	-2.521	2.426	-7.603	-2.133	.316	.828
-3.911	-4.438	21.438	-1.119	7.151	3.460	15.158	1.465
-4.352							
43.879	51.375	31.128	12.112	2.256	-1.346	-8.547	-3.845
4.129	-5.425	-16.417	-1.846	24.667	17.899	2.609	-3.067
-8.386	-14.665	-10.045	-10.764	-11.983	4.716	3.566	3.446
-2.413							
58.431	34.007	35.945	26.643	39.059	24.360	3.426	-33.840
4.996	6.491	-22.534	-20.909	-6.099	30.371	32.751	24.162
14.677	-16.127	-19.047	-18.231	-22.277	-20.150	.947	-11.411
14.564							
61.233	52.668	32.912	45.215	36.464	24.167	-3.063	.058
-12.660	3.867	.404	-3.385	-15.245	-19.840	-16.693	-18.238
-22.334	-28.625	-28.221	-27.091	-8.281	-3.325	5.462	9.562
18.113							
39.816	21.248	21.280	29.995	34.409	24.296	16.914	-7.243
-15.180	-17.255	-18.323	-12.143	-22.305	-27.670	-38.251	-31.155
-27.348	-25.522	-22.898	-19.813	-3.522	17.431	19.629	39.105
38.786							
26.750	-10.137	.591	14.126	26.799	29.073	22.600	23.350
12.290	-24.165	-23.607	-23.812	-19.703	-17.995	-28.969	-11.484
-20.000	-19.497	-15.501	-6.382	2.926	1.470	22.816	33.065
33.807							
-14.399	-3.812	-16.440	-6.846	7.445	18.374	21.679	24.337
10.580	-15.044	.867	2.124	14.035	14.520	-25.474	-30.646
-8.547	-8.639	-1.174	5.523	4.740	2.578	5.839	12.092
-11.603							
-32.714	-33.552	-26.327	-17.395	-16.759	9.812	20.577	12.111
-3.306	-17.522	25.716	34.329	34.126	40.127	23.902	-31.410
4.619	-1.089	-7.121	-5.500	-1.462	-3.978	.424	-5.246
-14.730							

B-51

-59.458	-55.122	-43.643	-33.676	-23.247	-9.275	1.742	5.792
-11.496	5.597	16.862	43.247	52.005	51.043	41.253	11.633
-24.255	.002	.366	1.750	1.098	-.398	-4.433	-4.435
-3.086							
PAVEMENT	CONSTRUCTION	COST FACTOR	NORTH			25	33 2
5.283	5.281	5.288	5.357	5.385	5.697	5.358	5.300
5.474	5.474	5.467	5.728	5.507	5.707	6.958	5.770
5.300	5.481	4.892	2.413	5.900	1.965	1.300	1.892
1.747							
5.219	5.224	5.289	5.382	5.749	5.772	5.300	5.261
5.545	5.611	5.576	5.736	5.615	7.199	7.411	5.761
8.000	5.288	4.496	2.537	2.252	1.919	1.300	1.819
2.003							
5.200	5.276	5.200	5.458	5.900	5.300	5.900	5.273
5.718	5.300	5.540	5.840	5.939	8.000	5.732	5.674
5.300	4.861	1.500	3.288	7.236	2.802	4.661	2.282
2.325							
5.200	5.300	7.179	5.715	7.188	7.023	5.388	5.704
8.636	8.494	5.715	5.440	5.865	5.532	5.485	5.300
4.532	4.471	2.799	3.172	7.186	5.212	4.880	5.203
5.331							
7.280	7.441	7.374	7.283	5.646	6.949	5.798	6.397
8.673	6.185	5.300	5.728	5.759	5.385	5.158	4.558
2.221	2.795	3.084	3.214	2.641	1.500	4.876	5.235
5.331							
5.737	5.860	5.571	5.379	5.214	5.200	6.929	7.407
5.804	5.707	5.700	5.300	5.772	5.300	5.434	5.030
1.985	3.036	2.162	6.032	5.547	5.355	5.015	5.300
5.151							
5.606	5.439	5.284	5.277	5.213	5.272	5.162	5.245
5.328	5.300	5.420	5.426	5.434	5.900	5.466	5.300
7.094	8.000	2.297	2.590	5.221	1.500	5.900	1.300
5.073							
5.625	5.656	5.691	5.456	5.219	5.218	4.856	2.551
5.149	5.900	5.300	5.728	5.404	5.045	5.146	7.553
3.261	2.717	2.701	4.820	2.504	4.156	1.868	4.640
4.825							
5.807	5.960	7.486	5.840	5.481	5.300	4.705	5.200
5.406	5.728	5.689	5.376	5.092	4.726	2.922	1.300
2.070	4.971	4.945	1.300	5.300	5.092	5.300	5.403
5.109							
5.758	6.006	7.672	7.476	5.403	5.406	2.613	2.746
5.300	5.279	5.316	4.726	1.300	4.613	2.170	2.885
5.900	4.821	2.390	5.300	5.900	5.372	5.714	5.364
5.332							
5.495	7.122	7.437	6.964	5.251	5.154	5.300	4.992
5.294	5.295	5.037	4.741	5.900	3.174	5.900	2.465
2.467	2.125	1.500	4.399	5.133	5.383	5.348	5.342
5.338							
5.345	5.355	5.253	4.760	1.300	2.958	5.604	5.338
5.300	5.300	5.055	4.973	5.217	5.286	4.991	2.281
1.869	2.108	2.318	2.740	5.125	5.300	5.300	5.300
5.300							
5.079	5.200	5.083	1.300	4.684	5.300	8.184	5.835
5.300	5.342	5.021	5.047	5.035	4.921	5.036	4.444
2.034	2.016	5.300	4.788	5.330	5.338	5.300	5.328
5.328							
5.161	1.300	5.575	5.300	5.059	5.693	8.495	5.884
5.867	5.343	5.300	4.839	3.004	3.054	4.863	2.381
1.993	2.268	4.628	4.551	5.110	4.662	4.981	5.123
5.157							
2.388	4.820	7.371	5.300	5.668	5.200	5.300	8.104
5.300	4.969	5.202	3.000	1.300	4.619	4.576	1.500
5.300	4.434	4.065	2.512	2.555	5.900	1.300	5.064
5.551							
5.060	5.610	6.098	7.248	5.839	5.279	5.603	5.420
5.081	5.082	4.944	1.300	2.087	2.030	2.053	4.388
2.293	2.172	4.282	2.041	5.900	2.501	5.900	5.128
5.149							

B-52

5.425	5.725	5.572	5.472	5.683	5.144	5.397	5.900
2.615	2.229	2.299	2.247	1.845	2.233	4.704	5.097
4.877	2.114	5.900	1.838	2.317	2.331	5.195	5.431
5.418							
5.300	5.413	5.133	5.900	4.696	5.200	5.285	2.360
2.148	4.453	2.339	2.076	4.428	4.938	5.300	5.569
5.509	4.546	1.300	4.086	1.500	2.375	5.068	5.426
5.386							
5.900	2.423	3.315	3.000	2.278	2.171	1.972	1.300
1.969	2.604	4.955	4.720	4.966	5.413	5.553	8.000
5.771	5.300	2.388	5.148	5.426	5.301	5.900	5.202
5.773							
5.900	4.988	3.633	5.900	3.368	2.831	2.118	1.300
5.900	2.601	2.886	5.161	5.352	5.300	5.519	7.357
5.692	8.000	5.478	7.200	7.612	6.634	2.359	5.030
5.900							
2.352	2.471	5.026	2.918	2.536	2.976	4.738	5.043
5.110	4.864	5.183	5.900	5.577	5.268	5.346	7.048
8.000	7.660	7.645	6.362	5.509	2.528	1.300	4.816
5.169							
2.174	2.214	2.350	4.763	2.167	2.260	1.500	2.447
2.584	5.034	7.232	7.336	7.300	5.355	3.273	5.705
8.000	7.752	7.458	6.775	6.306	1.500	1.679	2.637
4.982							
1.837	2.540	2.162	1.908	1.909	4.512	4.516	2.731
4.930	4.626	7.014	7.046	6.877	5.132	3.035	7.141
7.503	6.204	5.483	4.931	2.614	3.817	1.500	2.268
4.633							
2.517	2.561	1.598	1.732	1.755	2.014	2.379	6.827
8.000	2.576	4.902	5.166	5.300	2.941	2.650	5.531
7.127	5.634	3.411	6.436	4.599	1.880	1.990	1.300
4.967							
1.831	1.710	1.704	2.232	1.500	2.382	2.175	2.275
5.012	2.445	8.050	6.785	7.615	7.795	2.969	3.141
3.384	5.293	5.273	6.757	3.153	2.206	2.124	2.447
5.324							
3.053	2.942	2.232	2.400	4.894	6.418	2.285	3.000
1.500	1.300	7.593	3.365	5.393	3.210	8.000	3.159
6.619	6.756	3.039	8.000	5.263	4.572	2.464	1.500
6.641							
1.500	4.776	5.009	6.248	3.937	8.000	3.000	4.906
4.757	4.807	5.500	4.625	4.838	5.219	5.260	3.250
5.200	3.098	3.200	7.203	7.343	1.500	4.332	5.371
5.184							
5.464	5.515	5.352	1.500	5.087	5.200	6.615	4.591
5.900	4.904	4.730	2.179	1.500	5.300	5.200	4.949
1.500	7.056	7.212	7.502	7.342	4.894	2.215	2.306
2.275							
5.422	5.597	5.715	6.800	7.171	5.797	7.082	1.500
2.401	2.579	6.213	2.570	2.708	6.923	8.000	8.000
2.519	6.660	1.500	5.200	5.784	5.074	4.686	4.878
4.634							
5.764	5.433	8.000	5.792	5.560	5.460	5.338	4.995
4.456	5.137	6.635	8.000	1.500	6.699	5.489	5.300
4.473	5.200	5.101	5.147	5.216	5.200	5.285	5.127
5.108							
5.737	5.897	5.642	5.530	5.440	5.421	5.285	5.286
5.055	5.300	5.945	6.199	6.793	5.096	5.900	2.614
4.665	5.068	5.225	5.277	5.279	5.292	5.146	5.131
5.174							
5.392	5.507	5.492	5.433	5.294	5.283	5.268	5.351
5.327	5.900	7.396	5.618	5.200	5.323	5.502	3.042
5.300	5.060	5.285	5.288	5.294	5.046	4.973	2.959
5.501							
5.473	5.431	5.333	5.419	5.300	5.263	5.255	5.720
5.808	5.200	5.594	5.546	5.402	5.286	5.173	5.447
5.056	5.200	5.277	5.300	5.322	5.131	5.300	5.186
5.084							

k)

EXAMPLE OUTPUT

```

-----
I      THE GCARS SYSTEM      I
I  PURDUE UNIVERSITY VERSION  I
-----

```

SEARCH ACTIVITY - PREPARATION OF THE UTILITY SURFACE.

THIS ANALYSIS IS FORTH

```

-----
I      I      I      I      I      I
I  WEIGHT  I      FACTORS UTILIZED  I
I      I      I      I      I      I
-----
I      I      I      I      I      I
I  1.00  I  EARTHWORK COST FACTOR  I
-----
I      I      I      I      I      I
I  1.00  I  PAVEMENT CONSTRUCTION COST FACTOR  I
-----

```

CONTOUR MAP OF THE CONSTRUCTED UTILITY SURFACE

MAP IS DEVELOPED FROM 825 GRIDDED VALUES, CONSISTING OF 25 COLUMNS BY 33 ROWS.

CONTOUR LEVELS CALCULATED

GRIDDED DATA

LINEAR INTERPOLATION WITHIN GRID SQUARES

OBSERVED VALUES

ROWS = 33 COLS = 25
 ZBAR = 5.370 /SIG = 2.288 MAXZ = 11.651 MINZ = .025

VALUES EMPLOYED

ZMIN = .025 /MAX = 11.651 CI = 1.163
 GRID = 5 XINC = .200 YINC = .333
 CON = 10 PS = 4 CULMAX = 121

CONTOUR SYMBOL TABLE IS AS FOLLOWS

(ONLY EVEN SYMBOLS PRINTED IN BODY OF MAP)

SYMBOL	VALUE
*	NO DATA
-	BELOW LOW
0	.0252 TO 1.1477
1	1.1477 TO 2.3503
2	2.3503 TO 3.5128
3	3.5128 TO 4.6754
4	4.6754 TO 5.8379
5	5.8379 TO 7.0004
6	7.0004 TO 8.1630
7	8.1630 TO 9.3255
8	9.3255 TO 10.4881
9	10.4881 TO 11.6506
+	11.6506 OR OVER

[illegible]

FREQUENCY TABLE		INTERVAL	UNITS	PERCENTAGE	CUMULATIVE
		NO DATA	0		
		BELOW LOW	0	0.0000	0.0000
.0252	TO	1.1877	246	2.0959	2.0959
1.1877	TO	2.3503	911	7.7618	9.8577
2.3503	TO	3.5128	1094	9.3210	19.1787
3.5128	TO	4.6754	1348	11.4850	30.6637
4.6754	TO	5.8379	2515	21.4280	52.0917
5.8379	TO	7.0004	3692	31.4561	83.5478
7.0004	TO	8.1630	1326	11.2976	94.8454
8.1630	TO	9.3255	514	4.3793	99.2247
9.3255	TO	10.4881	87	.7412	99.9659
10.4881	TO	11.6506	3	.0256	99.9915
11.6506	OR	OVER	1	.0085	100.0000
		TOTAL	11737	100.0000	100.0000

SEARCH ACTIVITY - GENERATION OF ALTERNATIVES.

REPEATED USE OF A MODIFIED VERSION OF THE BRITISH ROAD RESEARCH LABORATORY MINIMUM PATH ALGORITHM GENERATES AS MANY AS SEVEN ALTERNATIVES.

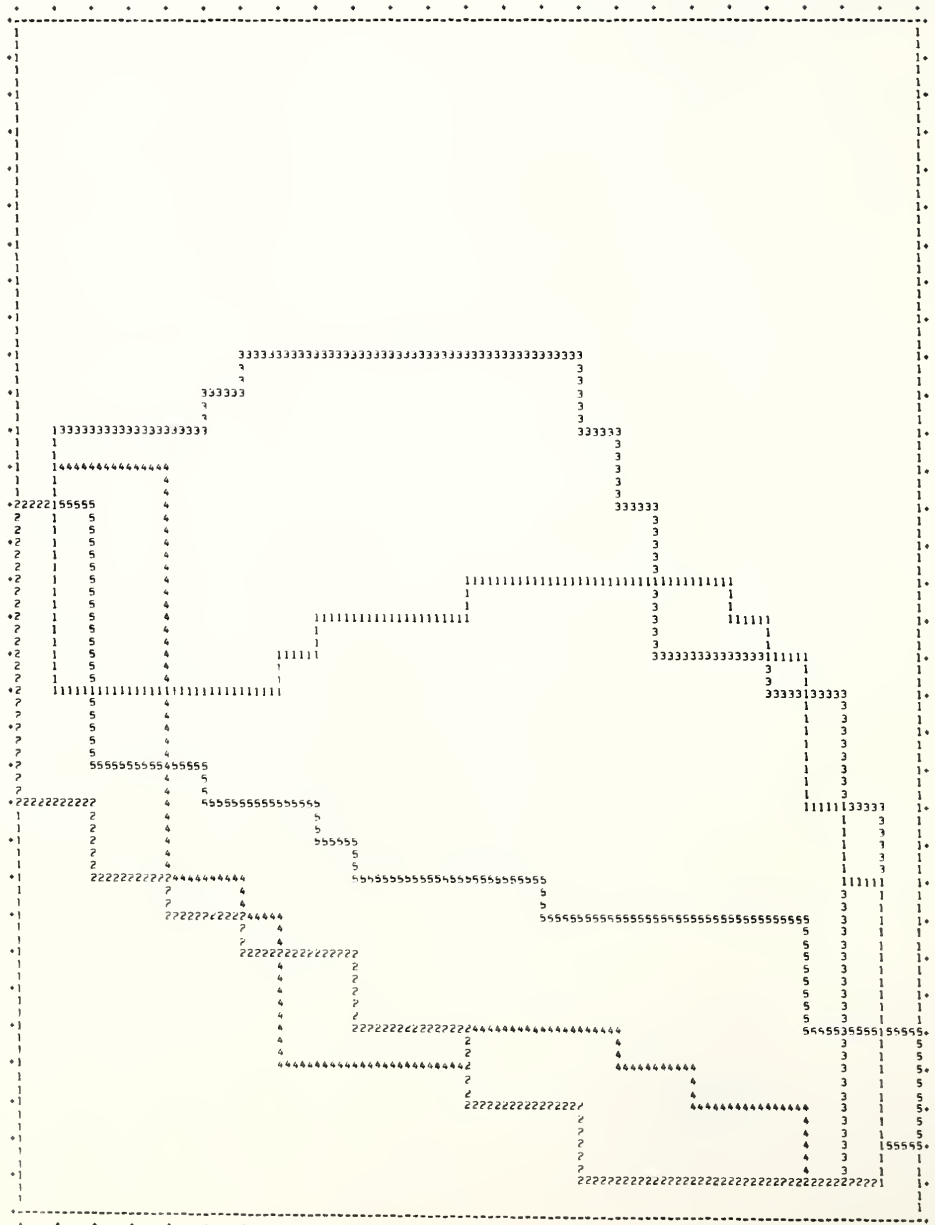
THE GENERATED ALTERNATIVES

CHOICE	PATH TOTAL	RELATIVE PATH VALUE	NUMBER OF LINKS
1	135.925	1.000	48
2	156.448	1.151	44
3	162.700	1.197	48
4	171.765	1.264	44
5	180.050	1.325	44

MAP OF GENERATED ALTERNATIVES.

4 CHOICES ARE SHOWN.
EACH IS LOCATED BY A STRING OF NUMBERS WHICH
CORRESPOND TO THE CHOICE NUMBER.

THE ALTERNATIVES ARE DEFINED ON A NETWORK
HAVING 25 COLUMNS BY 33 ROWS.



1) EXAMPLE OF PUNCHED OUTPUT

NUMBER	1 OF 2 FACTORS, WEIGHT= 1.00	2 OF 2 FACTORS, WEIGHT= 1.00	EARTHWORK COST FACTOR						
NUMBER	2 OF 2 FACTORS, WEIGHT= 1.00	COMMON UTILITY SURFACE.	PAVEMENT	CONSTRUCTION	COST FACTOR	COST FACTOR	25	33	2
5.745	5.736	5.448	5.610	5.799	6.207	5.792	5.905		
6.307	6.185	6.194	6.248	5.997	6.195	7.681	6.334		
5.675	6.240	5.697	2.210	7.524	1.820	1.026	1.437		
1.476									
5.950	5.809	5.722	5.659	6.287	6.201	5.667	5.864		
6.227	6.135	6.095	6.186	5.922	8.053	8.486	6.418		
9.951	6.254	5.318	2.532	1.924	1.446	.688	1.199		
.697									
5.968	6.212	6.062	5.669	6.443	5.431	6.413	5.620		
6.021	5.478	5.895	6.470	6.557	9.090	6.767	6.868		
6.526	5.549	1.162	3.471	8.641	2.962	5.319	1.719		
2.805									
5.686	5.911	8.103	6.119	8.180	7.803	5.687	6.000		
9.956	10.033	6.242	6.118	6.691	6.698	6.684	6.289		
5.245	4.797	3.225	7.984	5.437	5.823	5.823	6.793		
7.128									
8.194	8.516	8.685	8.284	6.027	7.745	6.104	7.046		
10.102	6.954	5.486	6.237	6.408	6.366	6.186	5.319		
1.329	2.123	3.186	2.906	1.975	1.616	6.313	6.113		
6.021									
6.230	6.310	6.062	5.553	5.807	5.818	7.832	8.296		
6.290	6.367	6.572	6.439	6.809	6.230	5.751	5.268		
1.188	2.628	2.388	7.086	7.428	6.168	6.483	6.598		
5.823									
6.399	6.032	5.631	5.555	5.701	5.650	5.396	5.438		
5.686	5.476	6.014	6.352	5.894	6.448	6.201	6.126		
8.852	10.222	2.074	3.533	6.886	1.420	7.701	.985		
5.595									
6.727	6.657	6.511	6.279	6.014	5.534	5.056	1.806		
5.351	6.625	6.009	6.062	5.598	6.088	5.797	9.750		
7.356	2.640	3.805	6.030	2.518	5.154	2.152	5.580		
5.539									
6.921	7.060	8.970	6.637	5.762	5.465	4.853	5.694		
5.855	6.322	6.487	6.393	5.861	5.019	3.450	.747		
2.058	6.666	4.964	.697	6.187	6.179	5.830	6.315		
5.582									
6.608	6.822	9.054	8.670	6.435	5.682	2.349	2.076		
6.096	6.508	6.493	5.886	.688	6.462	2.887	4.068		
8.016	6.055	2.024	6.346	7.250	6.416	7.007	6.266		
6.012									
6.069	8.064	8.629	8.349	6.074	5.750	5.426	5.753		
6.356	6.673	6.450	5.607	8.333	4.545	8.298	2.172		
2.889	1.753	.940	5.275	6.430	6.657	6.516	6.383		
5.966									
5.681	5.660	6.315	5.561	.575	2.339	6.247	6.599		
6.771	6.994	7.873	7.727	7.687	7.671	7.222	1.879		
.991	1.513	2.025	3.188	6.356	6.485	6.318	6.116		
5.849									
5.472	5.484	5.859	.234	5.043	6.164	9.814	7.665		
7.345	7.663	7.111	7.312	7.367	7.146	7.002	4.820		
1.067	1.388	6.292	5.941	6.547	6.345	5.861	5.927		
5.866									
6.292	.025	6.184	6.231	6.156	6.974	11.102	7.947		
8.491	7.862	7.394	6.870	4.387	3.948	6.166	2.676		
1.221	2.155	5.709	5.537	6.182	5.366	5.292	5.416		
5.497									
2.222	5.648	9.225	6.353	6.777	6.770	6.974	11.357		
6.940	7.361	7.316	4.477	1.879	5.774	5.090	1.392		
6.417	5.304	4.550	2.137	1.889	7.725	1.337	6.045		
5.778									
8.710	6.981	7.618	9.199	6.905	6.595	7.442	6.797		
6.440	6.882	6.832	1.534	1.694	1.048	1.884	4.257		
2.606	1.439	5.306	2.824	7.948	1.746	6.440	6.256		
5.502									
7.334	5.213	7.284	6.907	6.803	6.411	6.021	6.427		
1.857	1.527	3.543	2.292	.830	1.797	6.054	6.848		

6.456	2.432	8.041	1.068	1.630	2.627	6.510	5.816
5.860							
6.151	7.307	5.970	6.474	5.345	5.483	5.415	1.518
1.482	6.024	2.722	1.588	5.424	6.978	7.140	7.470
7.192	5.388	.707	4.590	.487	1.464	6.172	6.600
6.699							
7.118	3.044	3.594	3.511	1.958	1.579	1.419	.512
1.870	3.825	5.447	5.798	6.955	7.228	7.200	10.356
7.218	6.278	2.279	5.928	5.952	5.718	7.338	6.288
6.341							
6.444	6.546	5.385	8.670	5.094	4.593	2.308	1.509
8.260	3.994	3.140	6.424	6.777	6.595	7.014	9.417
6.839	9.846	6.131	8.336	9.098	7.620	2.645	5.516
7.224							
2.404	2.372	5.256	2.692	3.659	4.524	6.762	7.085
5.353	5.901	6.045	6.666	6.737	6.418	6.408	8.402
9.592	9.028	9.030	7.420	6.404	2.193	1.025	4.867
5.507							
2.235	2.025	2.230	5.954	3.098	2.784	.317	1.731
2.226	5.847	8.703	9.006	8.932	6.218	3.305	6.387
9.511	9.238	8.998	7.893	7.119	.512	.924	2.151
5.583							
2.054	2.478	1.789	1.240	.891	5.834	5.074	2.331
5.407	5.336	8.553	8.491	8.172	5.502	2.753	8.173
8.529	6.161	5.719	5.059	2.101	3.756	.524	1.949
5.551							
11.651	2.811	.988	.951	.794	1.145	2.768	8.907
9.564	1.945	5.332	5.817	5.635	2.327	2.029	5.769
8.136	5.927	2.949	7.020	4.629	1.003	1.269	.284
5.473							
2.073	1.409	1.030	1.361	.305	1.485	1.464	1.931
6.149	2.379	9.220	7.501	8.760	8.864	2.272	2.518
2.927	5.529	5.938	7.430	2.697	1.317	1.506	1.593
5.569							
3.503	3.545	2.062	1.803	4.932	6.976	1.555	2.404
.377	.139	8.956	2.848	6.184	3.050	9.154	2.600
7.429	7.776	2.632	9.363	5.682	4.559	1.670	.360
7.306							
1.770	5.587	5.952	7.395	4.578	9.712	2.394	5.759
4.817	4.923	6.274	5.046	4.955	6.094	6.211	3.264
5.666	2.852	3.081	8.474	8.767	.788	4.137	5.814
5.641							
7.218	7.068	6.340	1.431	6.074	5.909	7.287	4.465
6.564	4.987	4.662	1.279	.662	6.447	5.718	5.417
.844	8.541	8.742	9.107	8.407	4.883	1.381	1.610
1.787							
6.612	6.373	6.534	8.229	8.845	6.722	8.276	.457
1.883	2.177	7.133	2.034	2.482	8.336	10.068	9.886
2.355	7.924	.858	5.798	6.172	5.566	5.096	5.856
5.517							
6.741	5.866	9.102	6.455	6.465	6.388	6.056	5.610
4.596	5.824	7.841	9.648	.777	7.784	6.424	5.720
4.616	5.790	5.553	5.381	5.386	5.327	5.990	6.039
6.032							
6.387	6.333	6.311	5.913	5.809	6.060	5.961	6.030
5.364	5.811	6.322	6.699	7.810	5.521	6.892	2.569
4.783	5.127	5.328	5.536	5.518	5.480	5.366	5.506
5.552							
6.389	6.566	6.361	6.052	5.847	5.654	5.909	5.805
5.470	6.688	8.927	6.737	6.165	6.485	6.312	3.168
5.544	5.128	5.587	5.550	5.455	5.183	5.005	2.386
6.178							
7.185	7.017	6.589	6.449	6.021	5.613	5.409	6.143
6.409	5.433	6.256	6.869	6.897	6.715	6.311	5.977
5.716	5.290	5.403	5.470	5.483	5.206	5.539	5.513
5.211							

REFERENCES

1. Ayad, H., 1967, "System Evaluation by the Simplified Proportional Assignment Technique," Thesis Ph.D., Purdue University (unpublished), Lafayette, Indiana.
2. Martin, B. V., 1963, "Minimum Path Algorithms for Transportation Planning," Civil Engineering Systems Research Report R63-52, Massachusetts Institute of Technology, Cambridge, Massachusetts.
3. Turner, A. K., "Computer-Assisted Procedures to Generate and Evaluate Regional Highway Alternatives," Joint Highway Research Project, Final Report (Part I), December, No. 32, Purdue University, Lafayette, Indiana, 281p.
4. Turner, A. K., 1969, "The GCARS System FORTRAN IV Users Manual," Joint Highway Research Project, Final Report (Part II), September, No. 24, Purdue University, Lafayette, Indiana.
5. Turner, A. K., 1969, "The GCARS System FORTRAN IV Programmers Manual Part A - Programs for Data Collection and Evaluation," Joint Highway Research Project, Final Report (Part III-A), September, No. 25, Purdue University, Lafayette, Indiana, 104p.
6. Turner, A. K., 1969, "The GCARS System FORTRAN IV Programmers Manual Part C - Programs for Contour Mapping," Joint Highway Research Project, Final Report (Part III-C), September, No. 27, Purdue University, Lafayette, Indiana, 93p.
7. Turner, A. K. and Miles, R. D., 1969, "Computer-Aided Regional Highway Location Studies," Joint Highway Research Project Technical Paper, May, No. 12, Purdue University, Lafayette, Indiana, 39p.
8. Whiting, P. D., and Miller, J. A., "A Method for Finding the Shortest Route Through a Road Network," Operational Research Quarterly, Vol. 11, Nos. 1 and 2.

